Real Time Implementation of a Machine Learning Model Sign Language Recognition System Using Human Pose Estimation

Kristian Yair González Márquez

A Thesis Submitted in Partial Fulfilment

of the requirements for the

Degree of

Master of Science in Data Analytics



September 2023

Supervisor: David McQuaid

**CCT College Dublin**

**Assessment Cover Page**

|  |  |
| --- | --- |
| **Module Title:** | Capstone Project |
| **Assessment Title:** | Real Time Implementation of a Machine Learning Model Sign Language Recognition System Using Human Pose Estimation |
| **Lecturer Name:** | David McQuaid |
| **Student Full Name:** | Kristian Yair González Márquez |
| **Student Number:** | 2022421 |
| **Assessment Due Date:** | 22/09/2023 |
| **Date of Submission:** | 22/09/2023 |
| **Word Count:** | 7,028 |

**Declaration**

|  |
| --- |
| By submitting this assessment, I confirm that I have read the CCT policy on Academic Misconduct and understand the implications of submitting work that is not my own or does not appropriately reference material taken from a third party or other source. I declare it to be my own work and that all material from third parties has been appropriately referenced. I further confirm that this work has not previously been submitted for assessment by myself or someone else in CCT College Dublin or any other higher education institution. |

# ACKNOWLEDGEMENTS

**INDEX**

[FIGURES INDEX 4](#_Toc1)

[TABLES INDEX 4](#_Toc2)

[CHAPTER 1 INTRODUCTION 5](#_Toc3)

[Objectives 6](#_Toc4)

[Possible Elements of Validity to Apply in this Project 6](#_Toc5)

[CHAPTER 2 LITERATURE REVIEW 7](#_Toc6)

[2.1 Sign Language 9](#_Toc7)

[2.2 Object detection and object segmentation 10](#_Toc8)

[2.3 Human Pose Estimation 12](#_Toc9)

[2.4 Deep Neural Networks 15](#_Toc10)

[2.5 Activation Functions 17](#_Toc11)

[2.6 Sign Language Recognition Systems 25](#_Toc12)

[2.7 Challenges 28](#_Toc13)

[2.8 Conclusion 29](#_Toc14)

[CHAPTER 3 METHODOLOGY 30](#_Toc15)

[3.1 Ethical Considerations 30](#_Toc16)

[3.2 Primary Research Method 31](#_Toc17)

[3.3 Sampling 32](#_Toc18)

[3.4 Dataset Description 32](#_Toc19)

[3.5 Exploratory Data Analysis 33](#_Toc20)

[3.6 Keypoint Collection 39](#_Toc21)

[3.7 Neural Network Design 41](#_Toc22)

[CHAPTER 4 RESULTS 44](#_Toc23)

[4.1 Best Models 44](#_Toc24)

[4.2 Real Time Implementation 44](#_Toc25)

[4.3 Real Time Tests 44](#_Toc26)

[CHAPTER 5 CONCLUSIONS 44](#_Toc27)

[REFERENCES. 46](#_Toc28)

FIGURES INDEX

[Figure 1.1 Tanh Activation Function Representation 19](#_Toc1)

[Figure 1.2 ReLU Activation Function Representation 20](#_Toc2)

[Figure 1.3 Leaky ReLU Activation Function Representation 21](#_Toc3)

[Figure 1.4 ELU Activation Function Representation 22](#_Toc4)

[Figure 1.5 GELU Activation Function Representation 23](#_Toc5)

[Figure 1.6 SELU Activation Function Representation 24](#_Toc6)

[Figure 1.7 Mish Activation Function Representation 25](#_Toc7)

[Figure 1.8 Machine Translation methods distribution between 2004 and 2022. Taken from (De Coster et al., 2023) 27](#_Toc8)

[Figure 2.1 Frequency Data Distribution in Training Dataframe Top 50 Words 36](#_Toc9)

[Figure 2.2 Frames per Video in Training and Testing Dataframes Top 50 Words 38](#_Toc10)

[Figure 2.3 Frequency Data Distribution in Final Training Dataframe 39](#_Toc11)

[Figure 2.4 Frames per Video in Final Training Dataframe 39](#_Toc12)

TABLES INDEX

[Table 2.1 Top 50 Glosses with more videos in training ASL Citizen Dataset 35](#_Toc1)

[Table 2.2 Descriptive statistics of Training and Testing Datasets combined 36](#_Toc2)

[Table 2.3 Models tuned with Keras Tuner 42](#_Toc3)

# INTRODUCTION

*Machine Learning algorithms for Signal Language Recognition have been receiving more interest in recent years because they will provide more opportunities and improve communication in case of success to the deaf and hard of hearing population.* *Machine learning and deep learning algorithms can be trained on labelled datasets of sign language gestures, allowing them to learn the patterns and variations in hand movements, gestures and signs*.

*Human Interpreters are used as a way for traditional sign language translation but this is time consuming, expensive, and there is not always someone available to carry it out. Because of that finding an accurate and efficient machine learning algorithm has started to be a big point of interest, also it is essential to mention that there are some systems that require to have a device attached to help with the interpretation, however, this is expensive and it is again in the same situation as Human Interpreters, they are not always available.*

*By developing this system considering only a camera as input, capturing the human joints position prior to the classification, is an interesting approach that has been spoken for a few years now, and working on this there is a good chance to make a meaningful contribution in the data analytics area while helping in the inclusion of people in the deaf, mute, and hard of hearing community. Furthermore, this area of study comprehending a diverse skill set, including computer vision, object recognition, data pre-processing, machine learning, and deep learning which could help to improve significantly my knowledge, experience and expertise in data analytics.*

Objectives

* Determine the impact of different factors on the performance of Sign Language recognition models, such as lighting conditions, camera angles, and variations in hand gestures, and develop techniques to mitigate these factors.
* Implement a machine learning model testing different techniques to recognize and translate American Sign Language.
* Evaluate the usability and effectiveness of the interface in real-world settings considering speed and accuracy of the classification when using commodity hardware.

Possible Elements of Validity to Apply in this Project

Current: A lot of research has been done on Sign Language Recognition using machine learning algorithms, however because of the nature of technology constantly evolving, a lot of that research is now outdated due to recent developments in visual recognition systems as well as Deep Learning algorithms, so with the rapid advancements in these areas, it is important to ensure that the research project is based on the most current and up-to-date information available, this can be done verifying multiple sources of information even including grey materials but with caution.

Reliable: Plenty of information about the latest developments in visual recognition with the newest techniques is done in GitHub repositories or social network media like YouTube without strong documentation to back up their ideas. Because of this verifying the trustworthiness of the source of information is essential to minimize the risk of falling into misinformation, also, documenting the sources used in this project can also help ensure the transparency of the results, backing up the ideas on those grey materials with other sources can also be a good way to mitigate this risk.

# LITERATURE REVIEW

Millions of deaf and people with difficulties hearing use sign language to communicate between each other and also with other people, communication with signs could be difficult for individuals who are not fluent in this complicated expressive language.

Developing a sign language recognition system has the potential to shorten the communication barrier by translating automatically sign language to spoken/written language. This can be done using computer vision techniques capturing and analysing the visual features of sign language, passing it through a machine learning algorithm to classify it and translate the signs.

Sign language recognition systems have started to be a special research subject in recent years due to the recent advancements in computer hardware and software, as well as sign language data sets that have been developed for training and testing these systems.

The current state of research focused on the previously used algorithms and techniques will be treated in this literature review to identify what can be done in this project differently in order to have the most accurate and efficient model possible taking in consideration that the final goal of this research project is the implementation in a real time situation.

The chosen topics for this literature review were selected carefully, considering their importance as background knowledge prior to starting the Data Analysis Project, the literature review begins with a summary of Sign Language, providing an essential understanding of this unique form of communication. By exploring the fundamental aspects of Sign Language, to have an overview about it before jumping into the creation of models without a clear direction on how they should work.

Next, the literature review encompasses object detection, a crucial concept to explore Human Pose Estimation as once the person and their body parts are accurately detected, the model can focus solely on analysing and interpreting those specific elements, effectively disregarding the background information present in the image.

Deep Learning was also touch as it is behind the frameworks for Human Pose Estimation and will be also the responsible to perform the classification of the Sign Language.

And finally, a quick review on challenges that people working on Sign Language Recognition systems has encountered in the past because by exploring these challenges, such as limited datasets or variability in sign language gestures, it can be considered while implementing various techniques and models in their own projects.

It is important to mention that as the Data Analysis Project progresses, the literature review may be subject to change in subsequent submissions to incorporate new findings, methodologies or insights obtained during the testing and implementation phases. This adaptability ensures that the project remains updated, aligned with current research trends, and ultimately enhances the overall quality and effectiveness of the project.

* 1. Sign Language

According to (WHO, 2023) 5% of the world population or 430 million people have a certain degree of hearing loss and nearly 2.5 billion people by 2050 will have this condition. Sign language is a structured set of hand gestures with specific meanings used by hearing impaired people to communicate in everyday life (Konstantinidis et al., 2018) that allows them to communicate between each other and with the rest of people in a non-verbal way, a deaf person is a person with a disability to hear and someone mute is who has a disability to communicate verbally. It's challenging to communicate with others as they can't speak or listen and there is where sign languages come in useful because allow people to communicate without using spoken language (Sharvani Srivastava, 2021b). Sign Language is a structured form of hand gestures involving visual motions and signs to achieve the communication involving head, facial expressions, arms, hands, body, and fingers to represent the words (Cheok et al., 2019). There are several categories of hand gestures, including conversational gestures, controlling gestures, manipulative gestures, and communicative gestures (Ying Wu and Huang, 1999). Sign Language has its own vocabulary and grammar that is completely different from spoken and written languages, for transmitting information spoken languages employ the oratory abilities to create sounds that are mapped against certain words and grammatical combinations. The oratory parts are next absorbed and processed by the auditory faculties using visual senses (Sahoo et al., 2014).

Since studies in the 1960s and 1970s established that linguistic processes were not limited to the spoken modality, sign language study has evolved at an exponential rate (Stokoe, 2005), (Klima and Bellugi, 1979), over the course of four decades, studies of sign language have made substantial contributions to our knowledge of language, thought, and social interaction. Sign language is utilised in different regions of the world in the same way that spoken languages are, there exist different sign languages over the world some examples of these are the Japanese, British, Indian, Arabic, and American sign language (Zeshan, 2006). The most common visual language used by the deaf community in North America was the focus of several of these investigations which is American Sign Language (ASL) (Hauser et al., 2016).

ASL is a complete natural language with a number of linguistic features that set it apart from English (Liddell, 2003), some have speculated that ASL developed more than 200 years ago as a result of the blending of indigenous sign languages with French Sign Language (LSF, or Langue des Signes Française) (Cagle, 2010). Some features of LSF and the original local sign languages have made their way into modern American Sign Language, which is a rich, complex, and mature language. The current versions of ASL and LSF are two separate languages. Although they share some common indicators, neither set of users is capable of understanding the other's meanings (NIDCD, 2021), (Hosain et al., 2020) noted that about 6000 hand gestures are used to represent common words in American Sign Language while finger spelling is used to represent less common words and proper nouns.

American Sign Language is one of the most influential, as a result of their introduction in newly emerging educational systems, some sign languages have a high percentage of (old) ASL (-influenced) lexicon (Kusters, 2020). Not just in situations where the use of a national sign language was not yet institutionalized ASL was imported, the status of ASL and/or the accessibility of resources in ASL are likely significant factors in these shifts. Teachers in some deaf schools initially started out using a local or national sign language before switching to ASL (Kusters, 2021).

Signs can be written down in textual form using glosses. Glossing means choosing an appropriate English word for signs in order to write them down, although it is not the same as translation, it is conceptually very similar. A gloss on a signed story can be a series of English words written in small capital letters that correspond to the signs in an American Sign Language story (Othman and Jemni, 2012), for example, children who are fluent in American Sign Language and have a need to go through the process of learning to read English can benefit from using ASL gloss, which can be thought as the "elusive" intermediary system (Supalla et al., 2017), this SL gloss annotation form is crucial to help machines understanding and processing the sign language (Johnston, 2008).

* 1. Object detection and object segmentation

(Brownlee, 2019) refers to Object detection as a set of computer vision tasks related between each other involving the identification of objects in digital photographs. Basically, these tasks, can be divided in two major tasks, image classification, that is the class prediction of an object in an image, and object localization that is, as it name defines, locating one or more objects in an image, usually drawing a bounding box around them.

These object detection models can be classified in two major types: a two stage detector that locates the regions in which objects are present in the first one, and then on the second stage the object classification is done, in the other hand, for the one stage detector, it feeds the entire image to a CNN in one single step to locate the regions of interests and classify all the objects in the image (Zhao et al., 2019). These models produced highly accurate results in less computational time and outperformed traditional machine learning algorithms due to the recent development of high-performance GPUs for computation, and large benchmark datasets availability for training models (Alzahrani and Al-Baity, 2023).

Image segmentation is the association pixels process in an image matching them with their respective object class labels. This process is used in a lot of industries such as healthcare, transportation, robotics, fashion, home, and tourism. (Wang et al., 2022), it is also defined as the process of dividing images into regions with different features and extracting the regions of interest (ROIs). Image segmentation presents two challenges: how to define "meaningful regions" because the uncertainty of visual perception and the diversity of human comprehension lead to a lack of a clear definition of the objects, making image segmentation an ill-posed problem trying to effectively represent the objects in an image (Yu et al., 2023).

Object detection is essential in many computer vision tasks for analysing and comprehending visual data due to its ability to identify and locate specific objects within a given image or video, object detection has a wide range of applications thanks to this. Classification algorithms benefit from object detection because it allows them to focus on the information they need while ignoring irrelevant background. In Sign Language Recognition for example, once the hands are identified, the algorithm can ignore the rest of the scene and focus solely on the hands, leading to more precise interpretations of the signs being made. This focused approach not only improves accuracy but also speeds up the training process, as the algorithm does not require data from as many diverse situations. By leveraging object recognition, the algorithm can target specific objects of interest, resulting in faster and more effective training (Sunmok Kim, 2018).

Object detection also extends beyond SLR systems. Object detection is crucial for human pose estimation because it plays a vital role localizing where the body joints area (Amadi and Agam, 2023). By utilizing robust person detectors and concentrating on joint detection within bounding box regions, the algorithm can bypass the challenges of dealing with large-scale changes in images and unnecessary information. This focused method makes it easier to estimate human poses, which improves accuracy and performance. The algorithm can save time and effort by focusing its resources and computational power on the relevant objects of interest and discarding everything else (Wang et al., 2020).

Nowadays, one of the most used object detection algorithms is YOLO which states for “You Only Look Once” and was first released by (Redmon et al., 2016), since its creation, as is an open-source framework different people and companies such as Meituan and Ultralytics have been involved in improving and developing these algorithms with different versions until the latest release of YOLOv8 (Terven and Cordova-Esparza, 2023) having multiple variations along its journey.

Overall, object detection is important because it provides algorithms with valuable context for making decisions based on what they see in an image, SLR systems benefit greatly from object detection because it helps them better understand and interpret visual data in a variety of ways, including by increasing classification accuracy, boosting training efficiency, and allowing for precise pose estimation.

* 1. Human Pose Estimation

Human Pose estimation is the process of finding human figures in pictures and videos and figuring out which joints (keypoints) are present (Moryossef et al., 2021). It is essential for giving machines the ability to see and understand people and their interactions (Cao et al., 2021), 3D human pose estimation is widely increasingly being used for lots of applications such as sports instruction, limb rehabilitation training, augmented reality, and intelligent security (Meng and Gao, 2021). Human Pose Estimation can also be used for action recognition, the idea behind this is to extract body joint locations (keypoints) and then use this as input to another algorithm/neural network to select visual features in space and time to perform the action classification (Luvizon et al., 2018).

Deep learning based top-down and bottom-up pose estimation are the two categories in which human estimation pose algorithms are divided (Gui and Luo, 2022).

Traditional top-down human pose estimation detects people in an image or video and draws a bounding box around them using object detection, and then a pose estimator extracts body keypoints from the bounding box. This simple method has drawbacks like high computational cost and runtime proportional to the number of people. Bottom up is the opposite to top down as it draws keypoints on the image and then uses part affinity maps to map it to different people in the image (Gojariya et al., 2021), the most significant benefits of utilizing these second method are the lightweight network and the accelerated processing times (Martinez et al., 2017).

OpenPose was the first real-time multi-person system to jointly detect human body, hand, facial, and foot keypoints. Developed by researchers from Carnegie Mellon University and maintained working by Yaadhav Raaj and Ginés Hidalgo and was the first known method to do a whole body pose estimation consisting of three different blocks, body-foot detection, hand detection, and face detection which combined together output a total of 135 keypoints divided as follows: 25 body keypoints, 70 face keypoints, and 40 hand keypoints (Martinez, 2019). Openpose's CNN architecture is multi-stage and is considered a bottom up model. A CNN analyses the picture, producing a set of feature maps that are sent into the first stage. The first stage generates a collection of PAFs (Part Affinity Fields) by concatenating the prediction with the original image features repeatedly to get improved predictions. Using the same iterative approach as the first stage, the second stage predicts confidence maps. Part association is assisted by PAFs, while part detection is assisted by confidence maps. Each stage is composed of up of numerous convolution blocks that are generated by three 3x3 convolutional kernels at the same time (Badiola-Bengoa and Mendez-Zorrilla, 2021).

Nowadays, there exist multiple pose estimation techniques.since OpenPose was launched in 2016 lots of other methods had been developed. In this literature review, the actual state of the art will be reviewed. The first one to be reviewed is MediaPipe, a framework developed by Google for creating pipelines that carry out inference over any type of sensory data building a perception pipeline. A perception pipeline processes data from the real world using sensors and algorithms to create digital representations that can be analysed and understood by a computer system, face detection and segmentation, hands detection, pose detection, and any kind of object recognition are examples of perception pipelines (Lugaresi et al., 2019). For tasks like object detection, face detection, hand tracking, and pose estimation, it offers a wide variety of pre-built models which are trained on sizeable, varied datasets that are relevant to the task at hand like the COCO dataset which contains over 200,000 images of people in various poses and activities. They serve as the skeleton of nodes, edges, or landmarks, tracking important points on various body parts and each coordinate point is normalized in three dimensions (Halder and Tayade, 2021).

For this project four MediaPipe models listed below are of particular interest.

* Hand Landmark Model: locates 21 hand-knuckle coordinates as keypoints within the identified hand regions. About 30K real-world images and several rendered synthetic hand models imposed over different backgrounds served as the model’s training data. A palm detection model and a hand landmarks detection model are included in the hand landmarker model package. First, the palm identification model locates hands inside the input image, and then the hand landmarks recognition model recognizes specific hand landmarks on the palm detection model's cropped hand image.
* Face Mesh Model: similar to the hand landmark model, it detects face landmarks and facial expressions in selfie-like images and videos, producing a total of 468 3D keypoints. It performs the task using two deep neural network models, a face location detector that operates on the entire image, and a 3D face landmark model that operates on those locations and uses regression to predict the 3D surface.
* Pose Landmark Model: locates human body landmarks in images using machine learning, and they can process either a single image or an ongoing stream of images. The model produces 33 body pose landmarks in image coordinates and three-dimensional (x, y, z) world coordinates.
* Holistic Landmark Model: it is no more than the combination of the three models described above allowing to analyse full body gestures, poses, and actions giving a total output of 543 landmarks (33 pose landmarks, 468 face landmarks, and 21 hand landmarks per hand) in real time.

More MediaPipe documentation is available at the official Google’s Developer MediaPipe site (Google, 2023).

Another method of particular interest is Real-Time Multi-Person Pose Estimation (RTMPose), which is one of the most recent developments in human pose estimation released in 2023. It is integrated to the MMPose Opensource Toolkit, which allows for an easy deployment of the model. The model is trained on the COCO-Wholebody dataset, which gives the model the ability to recognize 133 dense landmarks with 68 on the face, 42 on hands, and 23 on the body and feet (Jin et al., 2020). RTMPose uses a top-down approach, which tackles the usual slow inference problem associated with this technique by working in conjunction with high-speed object detection models, using CSPNeXt as the backbone for the model, and executing human detection every certain number of frames instead of executing the detection in every single frame, in general RTMPose has demonstrated a good balance of speed and accuracy achieving better results in terms of accuracy/inference speed balance for whole-body pose estimation than other algorithms like OpenPose (Jiang et al., 2023).

* 1. Deep Neural Networks

Artificial Neural Networks (ANNs) often known as neural networks are innovative systems and computational approaches for machine learning, knowledge demonstration, and eventually the application of acquired information to maximize the output responses of complex systems (Chen et al., 2019). Artificial neural networks are built in the same way as the human brain which has billions of neuron nodes connected between each other, each neuron has a cell body that processes information by transporting it to and from the brain (van Gerven and Bohte, 2017).

The system is formed of a lot number of highly linked processing components known as neurons (Walczak and Cerpa, 2003), which work together to solve problems and communicate information via synapses (electromagnetic connections in the human brain). The neurons are interconnected and organized into layers. The data is received by the input layer, and the final outcome is generated by the output layer with one or more secret layers placed between the two of them (Dastres and Soori, 2021).

Deep Neural Networks are neural networks that including the input and output layers have more than three layers of neurons. These layered representations are using models referred to as "neural networks," which are arranged into literal layers and stacked one on top of the other (Schmidhuber, 2015).

In the 21st century, deep learning emerged and offered a neural network language that was superior to Natural Language Processing (NLP) due to its ability to vectorize multiple features against a sequential dataset in contrast to NLP models significantly enhancing unsupervised learning (Md Asif Jalal, 2018). This area of machine learning makes use of several high level abstraction models, deep learning and computer vision advancements have already produced valuable products for audio manipulation, face recognition, and vehicle detection.

The Deep Learning Framework provides a huge advancement for designing, training, and validating neural networks. Since signs are static, we can concentrate on hand posture and placements using real-time recognition technology, which can operate more quickly and provide an accurate translation. Different approaches had been used in previous works to do the SL classification, two of which are Convolutional Neural Networks and Recurrent Neural Networks.

The CNN model is an important Neural Network component used widely, but not exclusively, for image recognition and classification during sign and face detection or recognition (Pathak and Maheshwari, 2019). CNN models are composed of neurons with weights and biases. Specific neurons receive input data in response to actions, and weighted sums take over, activating certain functions and producing certain outputs. The CNN models are frequently used in multi-channel images (Dhulipala, 2022). CNNs as ANNs are inspired by human nature, CNNs emerged from the study of human’s visual cortex and thanks to the advancements in computation and data availability in the last years, these Deep Neural Networks have achieved an amazing performance on image recognition tasks, the most basic architectures stack a few Convolutional Layers, Pooling Layers, and so on making the image smaller and smaller between layers, finally passing through a common neural network which layers are the fully connected layers, these ones are the ones in charge of generating the output (Géron, 2019), the convolutional layer determines the output of neurons connected to important local regions, the pooling layer reduces the number of parameters by performing down sampling along spatial dimensionality, and the fully connected layers perform the same tasks as any standard Artificial Neural Network, producing class scores from the activations to be used for classification (O’Shea and Nash, 2015).

The most well-known CNNs are 2 dimensional CNNs that are used in images, but there are other types of CNNs such as 1D-CNN. 2D-CNNs were designed to operate exclusively in 2D data, which is why they perfectly fit images being the input of a 2D-CNN (width, height, depth), and due to its success, this type of neural network was tried for other engineering applications that involved 1D data, however, its application to 1D data was not as good as other traditional machine learning algorithms as the data was necessarily needed to be transformed to 2D data which is not a straightforward process always depending on the application, this problem was present for 1D data until 1D-CNN appearance showing a lower computational complexity allowing 1D-CNNs to be trained in standard computer configurations with no need of special hardware. As 2D-CNNs are mostly used for computer vision/images, 1D-CNNs are mostly used for sequential or time series data. In 2D-CNNs, the kernel slides in two dimensions, the width and height of the image being the dimensions it moves on, and the depth of the kernel is always the image's depth (number of channels). In 1D-CNNs, the kernel slides over one dimension only, the width of the kernel that can be customized, and the height of the kernel is fixed to the number of points per time step (Kiranyaz et al., 2021) .

Recurrent Neural Networks (RNNs) are supervised machine learning models that are composed of artificial neurons with one or more feedback loops, feedback loops are recurring cycles across time or sequence optimizing the network weights with the goal of minimizing the difference between the output and target pairs (i.e., the loss value), the simplest RNN has three layers which are input, recurrent hidden layer (containing the feedback loops), and output layer (Salehinejad et al., 2017).

Long short-term memory (LSTM) is a kind of RNN architecture that can remember values at arbitrary intervals. They are used to classify, process, and predict time series with known time lags and unknown durations. The LSTM is known as the cell state, and its recursive nature is shown by a looping arrow. As a result, the prior interval’s data is saved in the cell state. The cell state is adjusted by a remember vector situated beneath it, while the input modification gates adjust it. The gates also teach the network what to save, what to forget, what to remember, what to focus on, and what to output. The cell and hidden states are used to collect data for processing in the following state trying to resolve the gradient vanishing problem (Yamak and P. K. G, 2019) allowing to solve complex and artificial long time lag tasks (Hochreiter and Schmidhuber, 1997), additional to the LSTM units, there is another approach introduced a couple of years ago by (Cho et al., 2014) which name is Gated Recurrent Unit (GRU) consisting of an Update gate and a Reset Gate, being the Update gate an assistant to the model in determining how much past knowledge (from earlier time steps) should be passed on to the future, and the Reset Gate helps the model to decide how much past information has to be forgotten (Chung et al., 2014), LSTM and GRU can also be combined to achieve higher accuracies and this has been demonstrated by (Kothadiya et al., 2022) where six different architectures were tested LSTM, GRU, LSTM-LSTM, GRU-GRU, LSTM-GRU, and GRU-LSTM being LSTM-GRU the one achieving the highest accuracy predicting Indian Sign Language video frames over 11 signs.

* 1. Activation Functions

Activation functions, also known as transfer functions, are mathematical functions that determine whether a neuron is activated or not adding the ANNs to model nonlinear problems (Wiley, 2016), even when there is no evidence of having something similar in the human brain, it is the way to activate artificial neurons as biological neurons are activated (Montesinos López et al., 2022). There exist a lot of different activation functions that have been developed being the sigmoid function the first one to be used, and has been replaced over the years. As there is plenty of activation functions nowadays, the ones discussed in this literature review are some of the most popular, and that will be implemented in the neural network design process of this project.

There exist a lot of different activation functions that have been developed being the sigmoid function the first one to be used, and has been replaced over the years. As there is plenty of activation functions nowadays, the ones discussed in this literature review are some of the most popular, and that will be implemented in the neural network design process of this project.

Softmax: The softmax activation function is used for multi-label classification usually in the output layer of the network returning the probability distribution over output classes summing between all classes 1, a strong prediction will be a vector close to 1, leaving the rest close to 0, a weak prediction will have multiple possible categories with similar probabilities between each other (Montesinos López et al., 2022).

Tanh: Tanh goes from -1 to 1 having the advantage against sigmoid function to deal better with negative values(Patterson and Gibson, 2017) and is represented the same as the trigonometric function. Tanh is extensively used nowadays in RNNs such as LSTMs/GRUs as gates, tanh mathematical function is represented as:

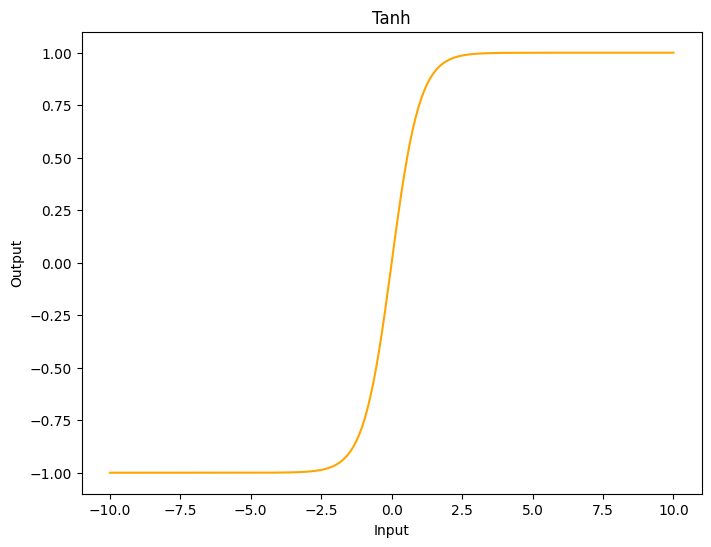


Figure 1.1 Tanh Activation Function Representation

Rectified Linear Unit: commonly known as ReLU, It is the go to activation function nowadays being one of the keys to deep learning success in recent years as they work in different situations, ReLU only activates if the input is above a certain number. When the input is less than zero, the output is zero; however, when the input is above a particular threshold, it has a linear relationship with the dependant variable and it is represented as (Patterson and Gibson, 2017):

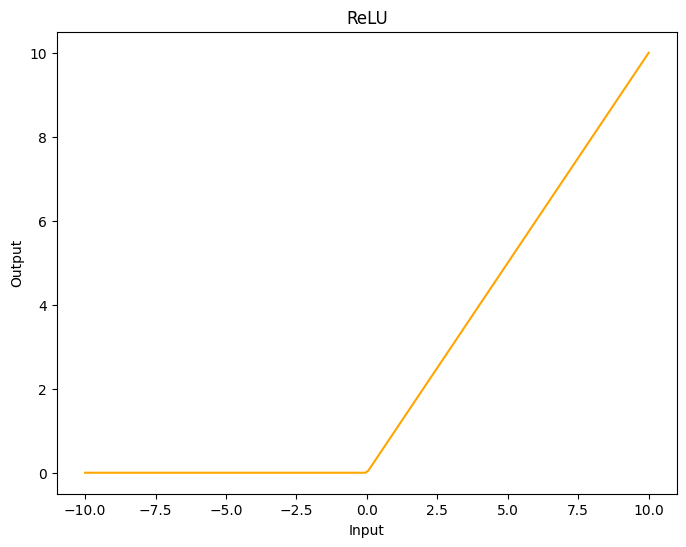


Figure 1.2 ReLU Activation Function Representation

Even when ReLU is the state of the art activation functions and have been demonstrated to train better than sigmoid/tanh activation functions. it has its downsides, the most important one is because the gradient of a ReLU is either zero or a constant, it is not easy to control the vanishing exploding gradient issue and this is known as “dying ReLU” (Montesinos López et al., 2022) which has been tried to be solved with several ReLU variants, some of them will be touched in this project.

Leaky ReLU: LeakyReLU is a ReLU variant that attempts to solve the dying ReLU problem by producing none zero output for negative input with a non-zero slope for negative values predefined between 0 and 1 enabling part of the negative feature information to be retained (Zhang et al., 2017), Leaky ReLU is represented as:

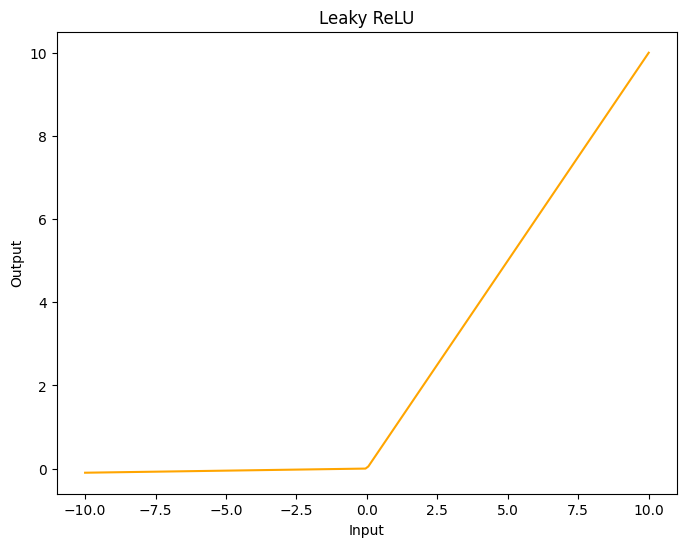


Figure 1.3 Leaky ReLU Activation Function Representation

Exponential Linear Units: ELUs in difference to ReLU have negative values which pushes the mean closer to zero which enables a faster learning, it is proved to produce better results than ReLU as the nonzero cofficient applied for negative values also helps the network to mitigate the “dying ReLU” problem. The only problem with ELU is that it requires more computation resources as it uses an exponential function making it slower than ReLU which is compensated when training by the faster learning, but in real time, it will always be slower than ReLU and other of its variants(Clevert et al., 2015). ELU is represented as

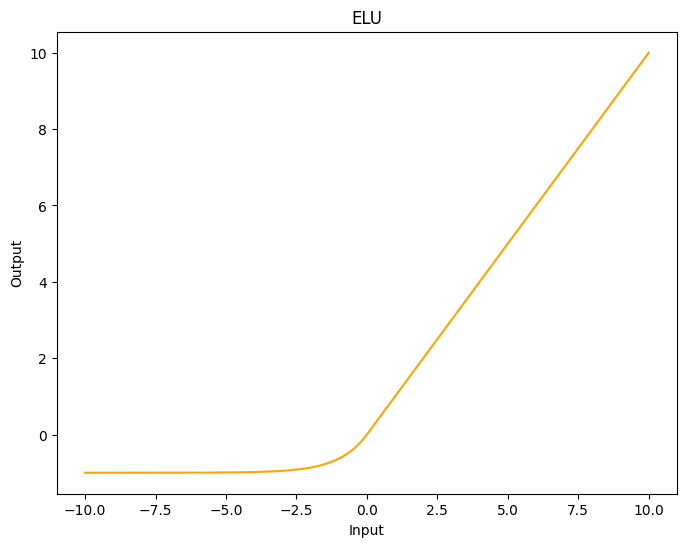


Figure 1.4 ELU Activation Function Representation

Gaussian Error Linear Units: GELU is designed based on the idea of combining regularization with RELU, it does a completely different job to weight its inputs in comparison to RELU instead of weighting the input based on its sign, it weights it based on the value, this is achieved by multiplying the input with the cumulative distribution function of the normal distribution at this input. As GELU is a computational intensive function because of the CDF calcuation, there is an approximation alternative which has only a slight variation but is faster (Hendrycks and Gimpel, 2023). The two representations of GELU are:

Or its approximation

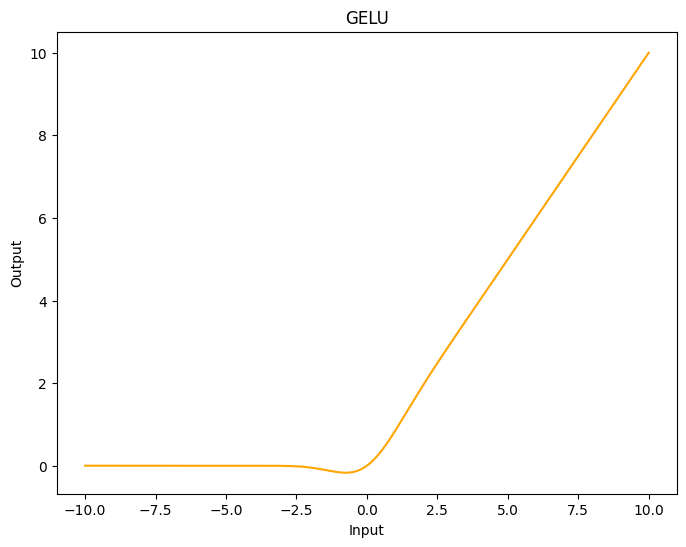


Figure 1.5 GELU Activation Function Representation

Scaled Exponential Linear Units: SELU is a variant of ELU attempting to bring fully connected neural networks again to the game bringing the concept of Self-Normalized Neural Networks (SNNs). SNNs are only formed with dense layers using the SELU activation function, causing the network to self-normalize, which means that during the training, the output of each layer will have a mean of 0 and a standard deviation of 1 addressing in this way the vanishing/exploding gradients problem. However, in order to this happen, certain conditions need to be met, such conditions are listed below:

* The input features must be standardized (mean 0 and standard deviation 1).
* Weights for every hidden layers have to be initialized with LeCun normal inizialization.
* The network’s architecture must be sequential.
* SELU activation with parameters λ ≈ 1.0507 and α ≈ 1.6733.
* Regularization with alpha-dropout, which is α and is the dropout technique also proposed with SNNs that randomly sets inputs to α value.

If these conditions are met, the self-normalization will occur and it is most likely that SELU will outperform other activation functions, if this conditions are not met, self-normalization is not guaranteed, which means it will not necesarilly outperform other activation functions (Klambauer et al., 2017). Even when the problem in this project, is not being addressed with fully connected neural networks, there are studies showing that SELU, has improved the performance on CNN-LSTM networks successfully (Phankokkruad and Wacharawichanant, 2019) (Huang et al., 2020).

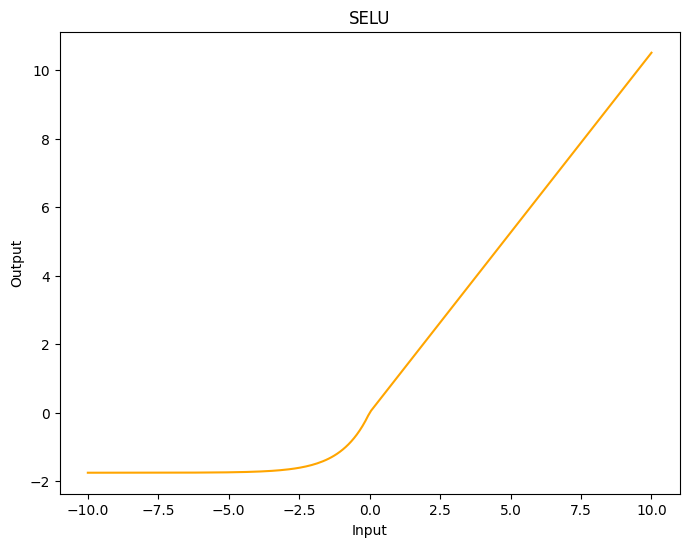


Figure 1.6 SELU Activation Function Representation

Mish: Mish is an activation function inspired by the activation function ‘Swish’ (Ramachandran et al., 2018) which was an activation function discovered using a novel automatic search technique and was tested in experiments designed for ReLU, just replacing ReLU with Swish increased the performance of the models, and Mish was found while studying the characteristics of Swish that makes it more effective than others. Mish trials were done in computer vision tasks with various standard architectures comparing its results against LeakyReLU, ReLU, and Swish being just outperformed by Swish in 2 tests. Mish is a smooth, continuous, self regularized, non-monotonic activation function mathematically represented as(Misra, 2020):

Or

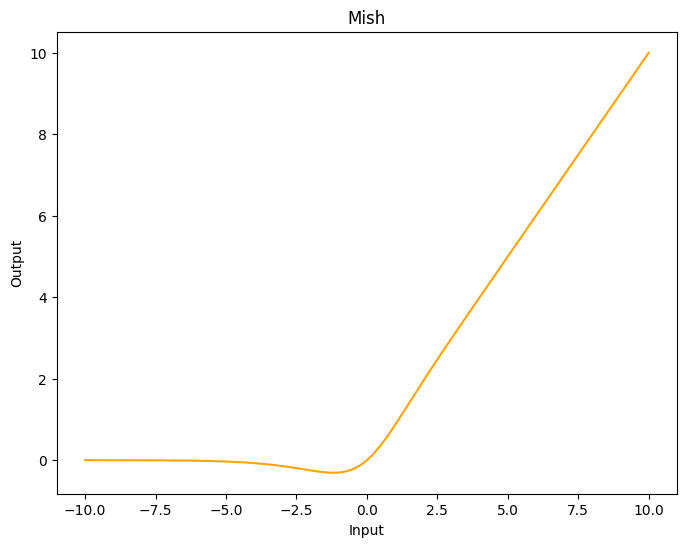


Figure 1.7 Mish Activation Function Representation

* 1. Sign Language Recognition Systems

Recent years have seen remarkable development in AI technologies related to Natural Spoken Languages, however, sign languages haven’t seen similar progress, especially in terms of word and sentence recognition even when sign language recognition technology has seen some progress over the years, with the advent of machine learning, particularly deep learning, more advanced recognition models have emerged but still not at the same level than spoken languages (Selvaraj et al., 2022). These models can learn and extract features automatically from large datasets, improving recognition accuracy, the use of machine learning techniques enables computerized systems to make decisions based on data and experience. Two datasets, a training dataset and a testing dataset, are required by the classification algorithms. The classifier learns from the training sets experiences, and the model is evaluated using the testing set (Rosero-Montalvo, 2018). Numerous authors have created effective methods for data collection and classification which can be divided into two categories based on the data acquisition method, these categories are direct measurement methods and vision-based approaches (Sarma and Bhuyan, 2021).

Direct measurement methods that use devices such as motion data gloves, motion capturing systems, or sensors (Kudrinko et al., 2021), basically any electronic system that can convert the signal, in this case Sign Language, into an electric or audible signal acting as a bridge for communication. However, there are some issues with sensor data acquisition, such as noise, poor human manipulation, and a faulty ground connection, as well as making the system more expensive because it requires specialized devices and training to use them adequately (Elakkiya, 2020). The extracted motion data can be used to track fingers, hands, and other body parts accurately, leading to the development of robust SLR methodologies (Sharvani Srivastava, 2021a), one of the most recognized devices of this kind is the Microsoft Kinect sensor, which captures a Red-Green-Blue (RGB) image and a depth map thanks to its infrared project, infrared sensor and RGB camera integration in one single device (Zhang, 2012), one example of success using the Kinect for American Sign Language recognition that was presented by (Cao Dong et al., 2015) where a Random Forest Classifier is fed with the joint angles in order classify 24 static signs reaching a 90% of accuracy.

The vision based SLR systems can work with a device as simple as a laptop webcam or phone camera getting RGB images to extract discriminative spatial and temporal information. Because it is not necessary to physically attach sensors to humans, vision based systems have recently gained popularity in recent years despite being prone to reliability issues in the past due to background noise, colours, and lighting variations in the real world (Kin Yun Lum, 2020). But nowadays with the latest developments in vision and machine learning, the classification of images has considerably improved, and with this, the vision based Signal Language Recognition algorithms. Sign Language recognition is not a new problem in computer vision, researchers have used classifiers from a variety of categories over the last two decades, which we can be roughly divided into linear classifiers, neural networks, and Bayesian networks (Anagha.G, 2022).

(De Coster et al., 2023) did recently an interesting article analysing the state of the art in Sign Language Recognition systems over the last years, the below table shows a selection of 57 research papers dedicated to SLR, and can be seen the evolution of them, from 2004 to 2018, all of them were using Statistical Machine Translation, but then, since 2019 all of them have started to use Neural Machine Translation, except for one article (Luqman and Mahmoud, 2020), where they used Rule-based Machine Translation.

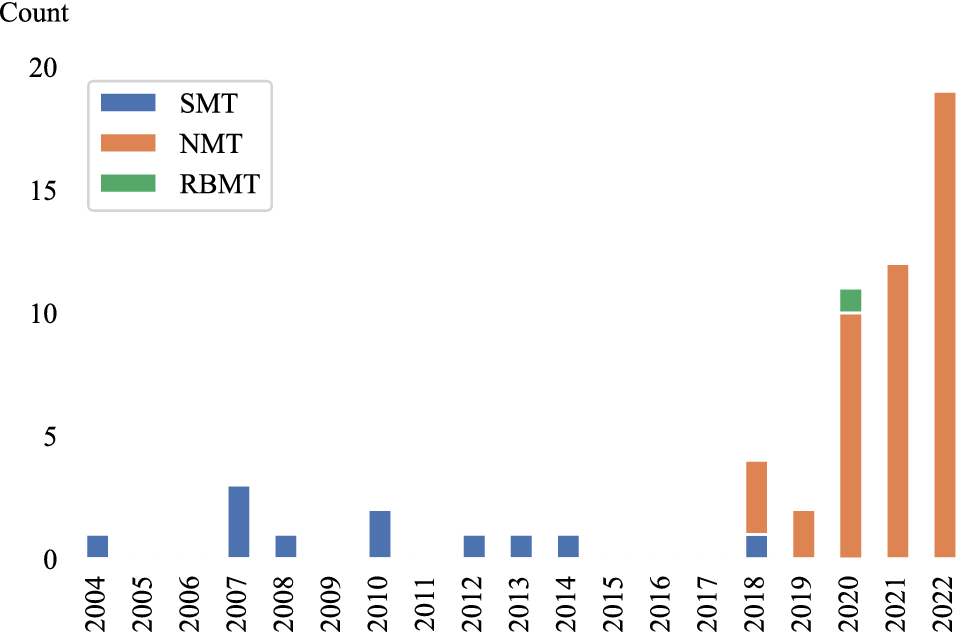


Figure 1.8 Machine Translation methods distribution between 2004 and 2022. Taken from (De Coster et al., 2023)

Another point to highlight is how the research for sign language, has increased since 2018 passing from only 10 since 2004, to 47 in the last years which corresponds with the trend of Neural Machine Translation systems.

Since OpenPose launched in 2016, different algorithms for pose estimation have been widely used trying to solve action recognition problems due to its simplicity and high accuracy to implement the different methods to do this, OpenPose and similar frameworks feed the vision based systems with that extra information about coordinates that the direct measurement systems have but with any kind of camera.

Human Body Pose Estimation is one of the fundamental tools used in the study of human behaviour, particularly in the areas of recognizing actions and sign languages. When there are a lot of occlusions or severe deformations, using the body features can help improve recognition accuracy (Rastgoo et al., 2021), this technique of human keypoints extraction allowed (Ko et al., 2018) to develop a novel Korean Sign Language recognition system that reached 89.5% accuracy feeding the keypoints to a neural network in 100 complete sentences that can be used in emergencies, another approach using keypoint extraction also was done recently by (Kim and Baek, 2023) where its model executes four principal tasks: extraction of keypoints, keypoint normalization, and skip sampling (SASS) to then achieve the Sign Language Translation, it is important to mention that they removed the lower body keypoints and face keypoints to end with a 55 keypoints extract which is after that normalized before feeding the algorithm. (Khan, 2022) after creating a system that detects finger spelling using MediaPipe Hand solution to extract hand only joints and OpenPose for a system able to recognize complete words extracting the full body keypoints concluded as a future work it might be interesting to try a combination between OpenPose and MediaPipe to improve the results being OpenPose the responsible to detect the body position and MediaPipe Hands module to extract the hand gestures keypoints.

* 1. Challenges

Sign Language Recognition systems faces the same challenges than every computer vision problem. The environment's structure, such as light and movement speed can have an influence with the predictive ability. The change in views causes the gesture to seem different in 2D space (Cheok et al., 2019), high computational cost is also an important point to mention even when Graphic Process Units in the last years have try to address this problem (Thompson et al., 2020), achieving high speed video real-time processing while maintaining accuracy is a difficult challenge which involves optimizing algorithms, hardware acceleration, and efficient memory usage. However, as stated by (Subramanian et al., 2022), finding a prototype that acquires the sign gesture and its corresponding text is the primary difficulty in creating a sign language recognition system and even when there are different techniques available, the challenges of hand tracking, occlusion of hand movements, high computational cost, feature selection and lower learning efficiency still exist.

(Singh, 2022) noted in three points, important facts that make sign language recognition systems challenging to be deployed nowadays which are “Limited number of datasets available”, “Domain restricted data”, “Lack of variety in datasets”, so data is an important challenge on this task, the author points that the few datasets available don’t have a wide variety of signers (10-20 average), lots of them are collected in the same environment, and have limited vocabulary what makes the model after training not being able to generalize.

The lack of standardization and availability of proper linguistic and grammar rules was also highlighted by (Attar et al., 2022) as an important obstacle to the development of effective sign language translation systems. Because of sign languages lack linguistic and grammar rules, researchers have been forced to use data-driven translation approaches in Sign Language Translation systems .

* 1. Conclusion

Machine Learning and deep neural networks are leading a significant improvement in the performance of sign language recognition systems. In this literature review the potential of deep learning to be implemented in an SLR has been explored. The ability of these networks to learn and extract features from large datasets leads to improved classification accuracy. It has been demonstrated that CNNs and RNNs can successfully be applied to the task of sign language recognition in several works before after extracting the human joints present on the image.

Object identification and segmentation techniques are really important for sign language recognition systems, these strategies aid in the identification and location of relevant items in the video stream, such as hands and body parts and further, estimating the signers pose is another crucial component of sign language recognition, it consists on finding and keeping track of the positions of key body landmarks within images or videos of the subject in question. Frameworks such as MediaPipe, OpenPose, and YOLOv8 are examples of some of the most popular options for pose estimation, however there are plenty of methods to extract keypoints nowadays.

Direct measurement methods and vision based approaches are the two primary classifications that can be applied to get information about sign language recognition systems. When it comes to capturing motion data, direct measurement methods involve the utilization of specialized hardware in the form of data gloves or motion capture systems. Vision based approaches rely on computer vision techniques and use RGB images obtained from cameras, this literature review has been helpful in identifying that the project will be done in a vision based method as it is more suitable in a real-time situation due to non-additional equipment being required other than a camera and a device to run the algorithm. When deciding the best approach, time and computational costs must also be considered to fit commodity hardware.

# METHODOLOGY

* 1. Ethical Considerations

It is important nowadays to consider ethics in every Artificial Intelligence project that will be developed as this is evolving rapidly and changing our lifestyles in positive and negative ways, (Green, 2018) made an important statement to take in consideration researching this science area, “AI, as the externalization of human intelligence, offers us in amplified form everything that humanity already is, both good and evil”, also one of the concerns the author points as the most important when dealing in this kind of applications is privacy, and privacy is a point that can be easily addressed on this project even if data is collected, as the final training dataset won’t be images or videos, the final dataset will consist of human pose joints coordinates only which will be just seen as numbers.

Despite the fact that machine translation systems have improved greatly in recent years, they are still sometimes not trustworthy enough for use in areas where lexical and conceptual precision are particularly important, such as in sectors dealing with cultural expression and literature or medical field. It is anticipated that AI-based Machine Translation will be developed principally for the primary world languages, especially English, because these languages have access to the massive datasets necessary for the technology's success. This is harmful to the preservation of different languages, because of this is always important to note that AI interpretations at this stage, can not substitute human interpreters and should be used with caution and in noncritical situations.

Considering these ethical challenges in sign language recognition projects not only safeguards individual privacy and fosters trust in AI technologies but also recognizes the importance of human expertise and cultural diversity. By incorporating ethical considerations into the development process, it is possible to create systems that respect user privacy, promote inclusivity, and enhance communication accessibility for individuals with hearing disabilities, while acknowledging the limitations and potential risks associated with AI-based translations.

* 1. Primary Research Method

The selected primary research method to be employed in this Sign Language Recognition project is experimentation. The experimentation will be focused on exploring three different frameworks for obtaining the pose estimation keypoints which are OpenPose, MediaPipe, and YOLOv8, also, different neural network architectures including CNN, CNN-LSTM, AND CNN-GRU will be evaluated to compare them and see which one can achieve the highest accuracy in real world settings.

The experimentation will be made in a systematically manner varying the combination of these techniques and architectures taking in consideration their influence on the accuracy of the sign language recognition system. By applying the principle of Concomitant Variation, which involves the manipulation of a variable and observing the resulting changes, the goal is to identify the optimal combination that yields the highest accuracy.

In each step of the experimentation, it is expected that temporal sequence will be present, the temporal sequence in this sign language recognition system will be related to the accuracy, as it will change accordingly, while different techniques of human pose extraction and architectures are tested and refined. The theoretical support will be provided by the literature review that will be the guidance while the experimentation process is conducted.

It is relevant to mention that this experimentation stage on the project won’t be restricted only to the findings mentioned on this literature review. If new relevant materials are presented while this project is worked, this new material may be included in the literature review, and in the experimentation. By doing this, it is ensured that the research remains current by incorporating the latest advancements and insights in the field.

If when the experimentation on this project is finalized, non-spurious association is not achieved, additional research methods may be incorporated, one of the possibilities could be conducting some depth interviews with experts or individuals with expertise in sign language recognition. Doing these interviews could provide insights to understand the factors influencing accuracy, explore further research needed, and identify different strategies or potential considerations that need to be addressed for this project.

* 1. Sampling

The populations of interest in this project are potential users of sign language translation and direct American Sign Language users which means people who interacts with mute, deaf and hard hearing community, and the mute, deaf and hard hearing community who sometimes suffer a gap of communication because there is not many people out of the deaf community that knows how to interpret sign language and an interpreter is not always available.

Collecting data directly from the deaf and hard of hearing community raises ethical considerations and requires careful attention to ensure the protection of their rights and privacy. Given the vulnerable nature of this population, alternative approaches will be adopted to address these concerns. Rather than getting data collected directly from individuals within the community, existing datasets available on the internet can be utilized for training and testing the sign language recognition models. These datasets, obtained from reliable sources, can provide valuable insights and information for developing and evaluating the models without directly involving vulnerable individuals.

For the experimentation phase, instead of sampling, each combination of the identified data acquisition technique (MediaPipe, RTMPose, CSPNext+UDP) and neural network architecture (CNN, CNN-GRU, CNN-LSTM) will be implemented and tested. This approach allows for a thorough evaluation of each combination performance and accuracy in real world settings. By systematically trying out each combination, the project aims to gather accurate and reliable data on the effectiveness of different techniques and architectures.

When executing the experimentation process, the achieved accuracies in real world settings for each of the combinations will be noted down by doing it in this way it is ensured that the project captures the practical implications and effectiveness of each combination, providing valuable insights for optimizing the sign language recognition models. By focusing on real world performance, the project aims to develop a robust system that can effectively bridge the communication gap and improve the overall accessibility and inclusivity for individuals using sign language.

* 1. Dataset Description

The data that have been used along the project to train the different models and testing is a portion of the just released “ASL Citizen” by (Desai et al., 2023) , it is a conjunction work of Microsoft Research, University of Washington, Boston University, and University of Maryland in an effort to support machine learning models for Isolated Sign Language Recognition (ISLR), it is composed of 83,399 videos for 2,731 different signs performed by 52 signers, 35 signers for training, 6 for validation, 11 for testing. It is important to mention that these videos were collected by people in everyday settings recording themselves with a webcam available, this is a great advantage comparing it to other previous ISLR datasets where the videos are recorded by professionals in laboratories or classrooms with certain characteristics as it gives the dataset a rich variety of conditions that allows the algorithm training to be more robust. The videos in the dataset where already cleaned by the research team removing the ones where there was not a person detected by the YOLOV3 algorithm, and also blurred the background in videos where additional people was also detected to protect their privacy.

The given dataset includes 3 different CSV files, each containing the videos for training, testing, and validation, the features contained are as below:

* Participant ID: Which of the 52 signers is performing.
* Video File: Name of the file.
* Gloss: The actual meaning of the file based on a previous released dataset ASL-LEX (Caselli et al., 2017).
* ASL-LEX Code: Gloss identifier in the ASL-LEX dataset.
  1. Exploratory Data Analysis

Taking in consideration the amount of time and resources available for this project only fifty words of the complete ASL Citizen catalogue are considered, and the decision on words to be used was based on the fifty glosses with more samples on the training dataset which can be found in Table 2.1.

| **Gloss** |
| --- |
| DOG1 |
| HURDLE/TRIP1 |
| BITE1 |
| BREAKFAST1 |
| DEMAND1 |
| DARK1 |
| MECHANIC1 |
| PARTY1 |
| DECIDE1 |
| WHATFOR1 |
| ROCKINGCHAIR1 |
| DEAF1 |
| EDIT1 |
| DEVELOP1 |
| RIVER1 |
| FINE1 |
| ELEVATOR1 |
| BELT1 |
| AXE1 |
| BACKPACK1 |
| SHAVE1 |
| CHRISTMAS1 |
| BEE1 |
| PATIENT2 |
| BASKETBALL1 |
| NOON1 |
| HALLOWEEN1 |
| LUNCH1 |
| EAT1 |
| TWINS1 |
| CANCER1 |
| DINNER1 |
| CONFUSED1 |
| RAZOR2 |
| MEAT1 |
| SQUEEZE |
| MICROSCOPE1 |
| DRAG1 |
| THEY1 |
| SINK |
| MOVIE1 |
| FLOAT1 |
| LOCK1 |
| DOWNSIZE1 |
| GUESS1 |
| KNIGHT1 |
| JEWELRY |
| MAPLE |
| FOREIGNER1 |
| HOSPITAL1 |

Table 2.1 Top 50 Glosses with more videos in training ASL Citizen Dataset

The frequency distribution on the training dataset once this filter was applied ends with 950 videos in total for 50 words, having a minimum of 18 videos and maximum of 24 videos per word falling 75% of them in the range of 18-20 samples.

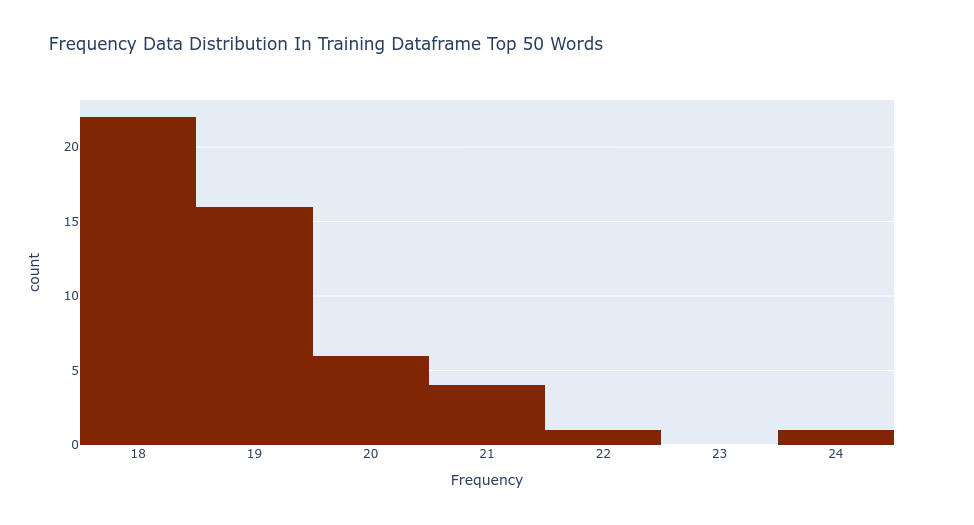


Figure 2.1 Frequency Data Distribution in Training Dataframe Top 50 Words

These same glosses where collected from the validation and test dataset ending with a total of 191 and 729 videos respectively. Taking in consideration that the end goal of this research is to train a neural network and the nature of the human pose estimation techniques used, some extra pieces of information not contained on the dataframes were gathered and analysed using the OpenCV library, such information becomes more important while the process of this work progresses, below is the list of columns added to the dataframes including a short description:

* Frames: Total number of the individual images that compose the video.
* FPS: Rate at which consecutive frames were recorded.
* Length: Total duration in seconds.
* Width: Number of pixels from the left edge to the right edge of the frame.
* Height: Number of pixels from the top edge to the bottom edge of the frame.

After this information was obtained the below statistics describe the combination of training and testing dataframes.

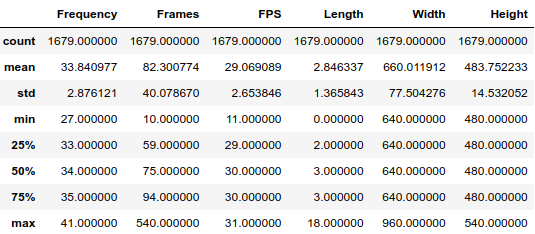


Table 2.2 Descriptive statistics of Training and Testing Datasets combined

From these, it can be seen how the videos were captured with different devices varying the FPS rate, which was expected from the start as Microsoft described the dataset was collected individually by the signers themselves without involving any kind of support other than the seed signer who indicated how the signs should be performed, which is helpful in general for the training, but a few more steps were required to be performed prior to starting collecting the keypoints data as the neural network input layer needs to have a specific input shape and there are videos with different lengths, meaning different quantity of frames.

The idea behind the keypoint collection is to have an array per video containing the keypoints per frame obtained from the Human Pose Estimation algorithm, because of this all the videos were filtered considering the ones lasting more than 1 second, with an FPS rate equal/greater than 29 and containing maximum 130 frames, this decision was made with the idea that the final arrays will be of 130 x number of keypoints, meaning that all the videos lasting less than 130 i.e. 120, the missing 10 keypoint arrays will be padded with zeroes. 130 is the number chosen as this is exact point between the 3rd quantile and the upper fence, ending with videos lasting more than a second and maximum an approximate of 4 seconds which is enough to capture the movement sequences to perform any sign, there are videos like “4256498922014438-SHAVE.mp4” lasting more than that where the signer performs a diversity of signs for the same action which will just provoke the input layer having to be larger, arrays for training bigger on memory containing zeroes only, and confusing the algorithm.

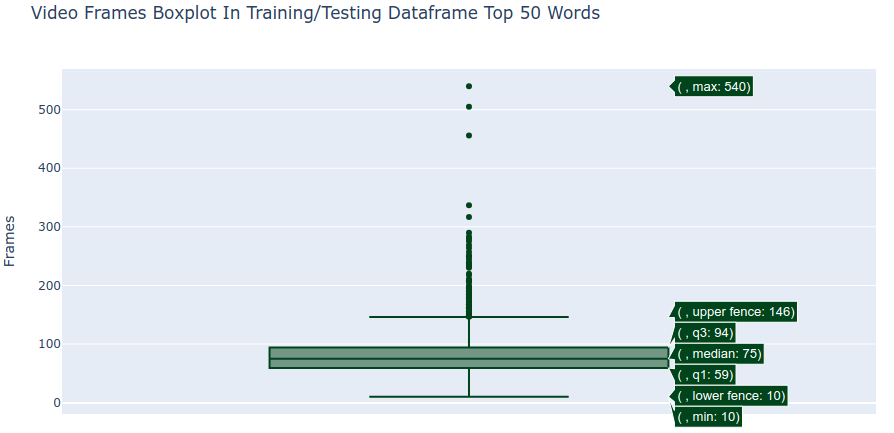


Figure 2.2 Frames per Video in Training and Testing Dataframes Top 50 Words

After the filtering has been applied, for training there were glosses with just 13 videos per gloss which was considered too low to train the network, because of this, the decision to join the training and testing datasets was taken ending with 1,338 videos, having a frequency distribution from 20 to 32 samples per gloss having a minimum of 45 frames and a maximum of 130 frames per observation, for validation the validation dataset was used applying the same rules ending with 161 videos having 2 to 5 samples per gloss.

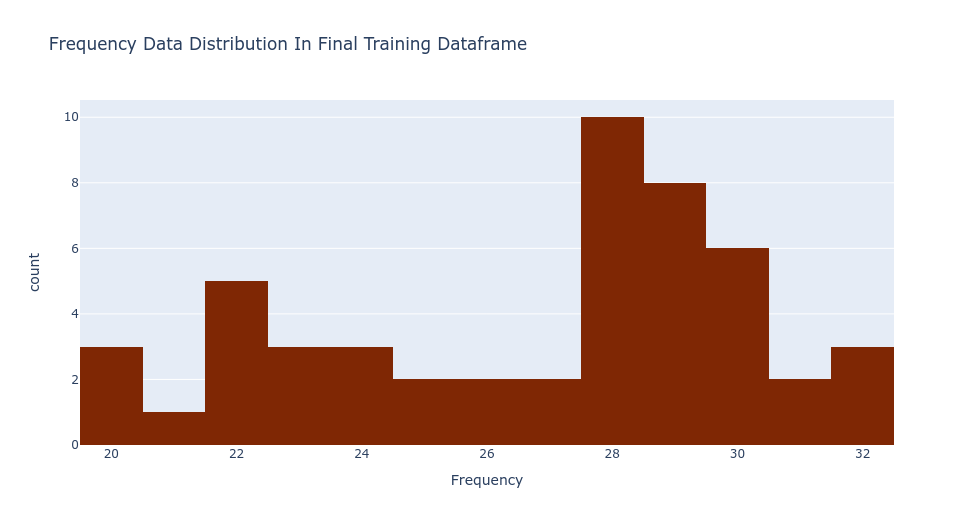


Figure 2.3 Frequency Data Distribution in Final Training Dataframe

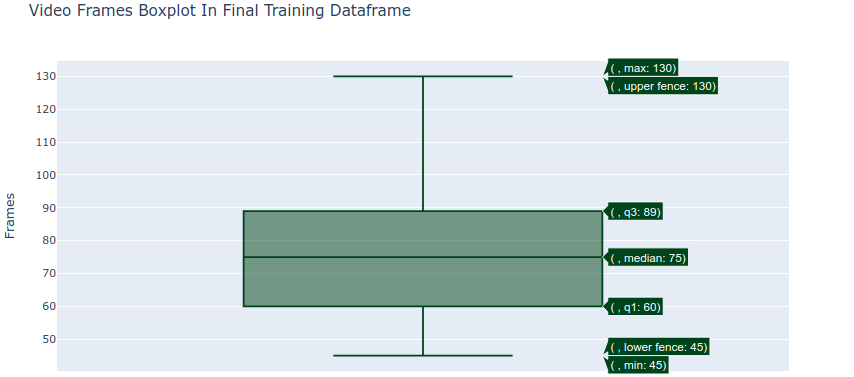


Figure 2.4 Frames per Video in Final Training Dataframe

* 1. Keypoint Collection

For this section, 2 main frameworks were used, Mediapipe and MMPose. Both of them were discussed in section 1.3, but briefly will be explained to have a better understanding.

Mediapipe, developed by Google is an open-source framework that allows to easily implement tasks like object detection, face detection, hand tracking, and pose estimation with pre-built models being the Holistic Model the one used on this project, the hands detection model was also tested, however it was discarded as the model at the moment this project was developed, the Mediapipe-hands instead of having a property identifying the left hand and right hand separately as in the Holistic model, it just creates an array with the keypoints visualized in the image, even though the right/left hands are identified, these keypoints are not always arranged in the same way pointing to additional challenges and noise for the algorithm to learn.

This being said, the Holistic model outputs 543 landmarks in total, 33 pose landmarks, 468 face landmarks, and 21 hand landmarks per hand. In the project only the 42 hand keypoints are considered, as using the whole given landmarks reaches a total of 1,662 keypoints (543 x 3 coordinates) increasing model complexity and noise induced by the proposed zero padding to standardize the inputs to 130, so 42 hand keypoints are flatten to get a total of 126 points and whenever a hand is not detected, all the points for the hand are automatically filled with zeroes ensuring that each frame has consistently 126 points. A loop running through each of the videos collects an array containing an array of 126 x number of frames per video, however this does not ensure the homogeneous shape required to feed a neural network and as mentioned previously pre-padding was the technique decided to feel the missing frames in videos shorter than 130, pre-padding was chosen over other techniques like post-padding or truncating because as mentioned by (Dwarampudi and Reddy, 2019), truncating drives to information loss and post-padding performs worse than pre-padding on LSTMs due to LSTMs nature to remember, so as in a human brain, it’s easier to remember recent information than older information, ending finally with a 3 dimension numpy array composed of (#videos, 130, 126), this process was also done flipping the frames 180 degrees prior to the keypoint collection using this as a data augmentation technique as overfitting was.identified in multiple occasions while designing the neural networks.

On the side of MMPose, it is a Pytorch-based pose estimation framework, part of the OpenMMLab project which allows to easily deploy algorithms to complete tasks like human pose estimation, animal pose estimation, face recognition, and fashion detection, the section involving the wholebody pose estimation algorithms is the one this project is focused on, specifically, CSPNext+UDP algorithm and RTMPose algorithm, CSPNext+UDP was decided to be tried as CSPNext is the backbone for RTMPose and when tested in the used hardware for the project, it works with a good accuracy and inference speed balance.

Both of these algorithms working under MMPose Inference API produce a list containing arrays with the image, 133 2D keypoints (17 for body, 6 for feet, 68 for face and 42 for hands), keypoints scores from 0 to 1, showing the prediction confidence for the 133 keypoints, and the classification boxscore. In the project the useful extracted information was the 133 keypoints with their respective scores, the scores are useful as this algortihms always give the 133 keypoints, so even if there is not i.e. a hand in the picture, the algorithm will still predict the keypoint but the confidence score will be zero, for the rest of the process. When the inferencer API outputs the keypoint coordinates x and y, it does it according to the coordinates of pixels depending the size of the image frame, it outputs different from a video with 640 x 480 resolution than a video 960 x 540 resolution, because of this, as previously identified 75% of the videos have a 640 x 480 resolution and all the videos, previous to be fed to the algorithm, are resized using openCV to standardize the keypoint coordinates outcomer. The rest of the process was the same than Mediapipe for each of the two algorithms flattening the keypoints and scores, and then filling an array of zeroes from right to left to do the pre-padding ending with a 3 dimension numpy array composed of (#videos, 130, 399), done as well flipping the frames 180 degrees prior to the keypoint collection for completing the data augmentation.

Finally, before passing the keypoints information for training, the keypoints collected were scaled for each of the three algorithms with the scikit-learn MaxAbsScaler, MaxAbsScaler was chosen among other scalers like MinMaxScaler or StandardScaler because the data is sparsed and sparsed data must not be centered, MaxAbsScaler scales each feature by dividing it with the largest maximum absolute value in each feature leaving transofrming the data to a [-1,1] range and its formula is:

* 1. Neural Network Design

When designing the neural network 3 different kind of layers were considered based on previous works identified in the literature review section, 1D-CNN, LSTM, and GRU layers as each of them have the ability to learn from temporal sequences, in this process, Keras tuner in conjunction with hyperband was used to adjust the hyperparameters of the network to find the optimal network configuration. Hyperband was chosen against other optimization algorithms because it is not as computational expensive as i.e. Bayesian Optimization because it is designed to allocate more resources to the most promising models, random sampling at the beginning and stop the bad performing combinations in an early stage, allowing to try a large number of combinations in less time which is a great point in favour considering the time limitation present on this project.

The models tried for each human pose estimation algorithm are presented in the table below.

| **Model** |
| --- |
| 1 1D-CNN Layer |
| 2 1D-CNN Layers |
| 3 1D-CNN Layers |
| 1 LSTM Layer |
| 2 LSTM Layers |
| 3 LSTM Layers |
| 1 1D-CNN + 1 GRU Layers |
| 2 1D-CNN + 1 GRU Layers |
| 1 1D-CNN + 1 LSTM Layers |
| 2 1D-CNN + 1 LSTM Layers |

Table 2.3 Models tuned with Keras Tuner

Even though the number of layers can be tuned with Keras Tuner, trying models adding layer by layer was preferred because it reduces the search space and gives the option to try the best model in each situation, as well as the option to select a smaller model if the inference time is too slow when evaluating the real time scenario, each model has been tuned to get the best validation F1 score possible, two executions per trial, early stop set to 30 epoch patience, and a maximum of 300 epoch. Each of the best models was trained until 500 epochs, with the weights that produce the best results saved to ensure squeezing the best out of each.

The models tried were designed with 1, 2, and 3 CNN/LSTM/GRU hidden layers and just 1 fully connected hidden layer in view of several studies showing that networks with more than 3 hidden layers are not optimal in terms of time complexity (Uzair and Jamil, 2020) and this is a crucial factor to consider in the evaluation on real time commodity hardware implementation, anyway the maximum number of 4 hidden layers was tried with the purpose of balancing the relation between the cost of the extra layer and the gained accuracy. 2 Dropout layers are also present being tuned with values from 0 to 0.9, if the chosen value by hyperband is 0, for the network it’s the same than if the Dropout layer did not exist.

The fully connected hidden layer present in every model was tuned with neurons ranging between 70 and 120, and the activation function, which selection is explored later. The tuned hyperparameters for the CNN Layers were the activation function, number of filters, searching values ranging from 50 to 200, 50 to maintain the relationship suggested in various literatures stating that the number of neurons should be a number between the total inputs and outputs, 200 as maximum, because it was observed that after 200, the training time increase was already significant (being more notable in methods with input sizes of 130,399). Kernel size is tested with sizes between 2 to 20 at different scales depending the model tried, and MaxPooling in the same range than the kernel size with the exception in the models composed by 3 1D-CNN Layers where the MaxPooling size was fixed to 2 because the combinations of sub-samplings in lots of sizes combinations resulted in errors due to the nature of MaxPooling causing to have a smaller input size than the number of down samples desired.

The LSTM/GRU layers were designed in the same range of the CNN layers for the number of neurons, adding L2 regularization to help reduce overfitting where is present, regularization was added to LSTM/GRU layers due to their trend to overfit data easily. The activation function is left with the default ‘Tanh’ as this the activation function required by CUDA/CUDNN in these kind of network layers.

For activation functions to be tested, for the Mediapipe algorithm the hyperparameter search was done using ‘ReLU’, ‘LeakyReLU’, ‘GELU’,’ELU’, and ‘SELU’, ‘MISH’ function was not included because is not implemented in Tensorflow it was out the scope at the beginning, however, after finding the way to include it in the training, trials substituting the activation function in the best model found were done in a attempt to compensate the situation, it was found that ‘MISH’, as expected, leads to improve some of the models accuracy. For the other two algorithms based in MMPose framework, the activation function selection was done with ‘ReLU’, ‘GELU’, ‘ELU’, ‘SELU’, and ‘MISH’, ‘LeakyReLU’, was removed as it was observed from MediaPipe that it could not outperform the others in any model.

# RESULTS

* 1. Best Models
  2. Real Time Implementation
  3. Real Time Tests

# CONCLUSIONS

REFERENCES.

Alzahrani, N. and Al-Baity, H.H. (2023) “Object Recognition System for the Visually Impaired: A Deep Learning Approach Using Arabic Annotation.” Electronics, 12(3), p. 541. DOI: 10.3390/electronics12030541.  
 Amadi, L. and Agam, G. (2023) “Weakly Supervised 2D Pose Adaptation and Body Part Segmentation for Concealed Object Detection.” Sensors (Basel, Switzerland), 23(4), p. 2005. DOI: 10.3390/s23042005.  
 Anagha.G, S.S.V.N. (2022) “Sign Language Recognition Using Machine Learning: A Survey.” International Journal of Innovative Science and Research Technology, 7(4), pp. 1211–1215.  
 Attar, R.K., Goyal, V. and Goyal, L. (2022) “Open Issues and Challenges in Automatic Sign Language Translation Systems.” AIP Conference Proceedings, 2455(1), p. 040003. DOI: 10.1063/5.0100903.  
 Badiola-Bengoa, A. and Mendez-Zorrilla, A. (2021) “A Systematic Review of the Application of Camera-Based Human Pose Estimation in the Field of Sport and Physical Exercise.” Sensors (Basel, Switzerland), 21(18), p. 5996. DOI: 10.3390/s21185996.  
 Brownlee, J. (2019) Deep Learning for Computer Vision: Image Classification, Object Detection, and Face Recognition in Python. Machine Learning Mastery.  
 Cagle, K.M. (2010) “EXPLORING THE ANCESTRAL ROOTS OF AMERICAN SIGN LANGUAGE: LEXICAL BORROWING FROM CISTERCIAN SIGN LANGUAGE AND FRENCH SIGN LANGUAGE.”  
 Cao Dong., Leu, M.C. and Yin, Z. (2015) “American Sign Language Alphabet Recognition Using Microsoft Kinect.” In 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Boston, MA, USA: IEEE, pp. 44–52. DOI: 10.1109/CVPRW.2015.7301347.  
 Cao, Z. et al. (2021) “OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields.” IEEE Transactions on Pattern Analysis and Machine Intelligence, 43(01), pp. 172–186. DOI: 10.1109/TPAMI.2019.2929257.  
 Caselli, N.K. et al. (2017) “ASL-LEX: A Lexical Database of American Sign Language.” Behavior Research Methods, 49(2), pp. 784–801. DOI: 10.3758/s13428-016-0742-0.  
 Chen, M. et al. (2019) “Artificial Neural Networks-Based Machine Learning for Wireless Networks: A Tutorial.” IEEE Communications Surveys & Tutorials, 21(4), pp. 3039–3071. DOI: 10.1109/COMST.2019.2926625.  
 Cheok, M.J., Omar, Z. and Jaward, M.H. (2019) “A Review of Hand Gesture and Sign Language Recognition Techniques.” International Journal of Machine Learning and Cybernetics, 10(1), pp. 131–153. DOI: 10.1007/s13042-017-0705-5.  
 Cho, K. et al. (2014) “Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation.” DOI: 10.3115/v1/D14-1179.  
 Chung, J. et al. (2014) “Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling.”  
 Clevert, D.-A., Unterthiner, T. and Hochreiter, S. (2015) “Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs).” Under Review of ICLR2016 (1997).  
 Dastres, R. and Soori, M. (2021) “Artificial Neural Network Systems.” International Journal of Imaging and Robotics, 21(2).  
 De Coster, M. et al. (2023) “Machine Translation from Signed to Spoken Languages: State of the Art and Challenges.” Universal Access in the Information Society. DOI: 10.1007/s10209-023-00992-1.  
 Desai, A. et al. (2023) (arXiv:2304.05934) Available at: http://arxiv.org/abs/2304.05934 (Accessed: August 19, 2023).  
 Dhulipala, S.A. (2022) “Sign and Human Action Detection Using Deep Learning.” Journal of Imaging.  
 Dwarampudi, M. and Reddy, N.V.S. (2019) (arXiv:1903.07288) Available at: http://arxiv.org/abs/1903.07288 (Accessed: August 24, 2023).  
 Elakkiya, R. (2020) “Machine Learning Based Sign Language Recognition: A Review and Its.” Journal of Ambient Intelligence and Humanized Computing.  
 Géron, A. (2019) Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. O’Reilly Media, Inc.  
 van Gerven, M. and Bohte, S. (2017) “Editorial: Artificial Neural Networks as Models of Neural Information Processing.” Frontiers in Computational Neuroscience, 11. Available at: https://www.frontiersin.org/articles/10.3389/fncom.2017.00114 (Accessed: May 11, 2023).  
 Gojariya, D. et al. (2021) “REVIEW OF LITERATURE SURVEY ON DIFFERENT HUMAN POSE ESTIMATION AND POSE COMPARISON TECHNIQUES.” 9(10).  
 Google. (2023) MediaPipe Solutions Guide | Google Developers. Available at: https://developers.google.com/mediapipe/solutions/guide (Accessed: May 4, 2023).  
 Green, B.P. (2018) “Ethical Reflections on Artificial Intelligence.” Scientia et Fides, 6(2), pp. 9–31. Available at: https://apcz.umk.pl/SetF/article/view/SetF.2018.015 (Accessed: May 16, 2023).  
 Gui, Z. and Luo, J. (2022) (arXiv:2212.11344) Available at: http://arxiv.org/abs/2212.11344 (Accessed: May 19, 2023).  
 Halder, A. and Tayade, A. (2021) “Real-Time Vernacular Sign Language Recognition Using MediaPipe and Machine Learning.” 2(5), pp. 9–17.  
 Hauser, P.C. et al. (2016) “American Sign Language Comprehension Test: A Tool for Sign Language Researchers.” The Journal of Deaf Studies and Deaf Education, 21(1), pp. 64–69. DOI: 10.1093/deafed/env051.  
 Hendrycks, D. and Gimpel, K. (2023) (arXiv:1606.08415) Available at: http://arxiv.org/abs/1606.08415 (Accessed: September 1, 2023).  
 Hochreiter, S. and Schmidhuber, J. (1997) “Long Short-Term Memory.” Neural Computation, 9, pp. 1735–80. DOI: 10.1162/neco.1997.9.8.1735.  
 Hosain, A.A. et al. (2020) “Body Pose and Deep Hand-Shape Feature Based American Sign Language Recognition.” In 2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA). 2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA). pp. 207–215. DOI: 10.1109/DSAA49011.2020.00033.  
 Huang, Z. et al. (2020) “SNDCNN: Self-Normalizing Deep CNNs with Scaled Exponential Linear Units for Speech Recognition.” In ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 6854–6858. DOI: 10.1109/ICASSP40776.2020.9053973.  
 Jiang, T. et al. (2023) (arXiv:2303.07399) Available at: http://arxiv.org/abs/2303.07399 (Accessed: September 4, 2023).  
 Jin, S. et al. (2020) “Whole-Body Human Pose Estimation in the Wild.” In Proceedings of the European Conference on Computer Vision (ECCV).  
 Johnston, T.A. (2008) “Corpus Linguistics and Signed Languages: Sixth International Language Representation and Evaluation Conference” O. Crasborn, E.E. and I. Zwitserlood, null (eds.). Proceedings of the Sixth International Language Representation and Evaluation Conference, pp. 82–87.  
 Khan, R. (2022) “Sign Language Recognition from a Webcam Video Stream.”  
 Kim, Y. and Baek, H. (2023) “Preprocessing for Keypoint-Based Sign Language Translation without Glosses.” Sensors, 23(6), p. 3231. DOI: 10.3390/s23063231.  
 Kin Yun Lum, Y.H.G.Y.B.L. (2020) “American Sign Language Recognition Based on MobileNetV2.” Advances in Science, Technology and Engineering Systems Journal, 5(6, pp. 481–488.  
 Kiranyaz, S. et al. (2021) “1D Convolutional Neural Networks and Applications: A Survey.” Mechanical Systems and Signal Processing, 151, p. 107398. DOI: https://doi.org/10.1016/j.ymssp.2020.107398.  
 Klambauer, G. et al. (2017) “Self-Normalizing Neural Networks.” In Advances in Neural Information Processing Systems. Curran Associates, Inc. Available at: https://proceedings.neurips.cc/paper\_files/paper/2017/hash/5d44ee6f2c3f71b73125876103c8f6c4-Abstract.html (Accessed: September 1, 2023).  
 Klima, E.S. and Bellugi, U. (1979) The Signs of Language. Harvard University Press.  
 Ko, S.-K., Son, J.G. and Jung, H. (2018) “Sign Language Recognition with Recurrent Neural Network Using Human Keypoint Detection.” In Proceedings of the 2018 Conference on Research in Adaptive and Convergent Systems. RACS ’18: International Conference on Research in Adaptive and Convergent Systems. Honolulu Hawaii: ACM, pp. 326–328. DOI: 10.1145/3264746.3264805.  
 Konstantinidis, D., Dimitropoulos, K. and Daras, P. (2018) “SIGN LANGUAGE RECOGNITION BASED ON HAND AND BODY SKELETAL DATA.” In 2018 - 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON). 2018 - 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON). Helsinki: IEEE, pp. 1–4. DOI: 10.1109/3DTV.2018.8478467.  
 Kothadiya, D. et al. (2022) “Deepsign: Sign Language Detection and Recognition Using Deep Learning.” Electronics, 11(11), p. 1780. DOI: 10.3390/electronics11111780.  
 Kudrinko, K. et al. (2021) “Wearable Sensor-Based Sign Language Recognition: A Comprehensive Review.” IEEE Reviews in Biomedical Engineering, 14, pp. 82–97. DOI: 10.1109/RBME.2020.3019769.  
 Kusters, A. (2021) “International Sign and American Sign Language as Different Types of Global Deaf Lingua Francas.” Sign Language Studies, 21(4), pp. 391–426. DOI: 10.1353/sls.2021.0005.  
 Kusters, A. (2020) “The Tipping Point: On the Use of Signs from American Sign Language in International Sign.” Language & Communication, 75, pp. 51–68. DOI: 10.1016/j.langcom.2020.06.004.  
 Liddell, S.K. (2003) Grammar, Gesture, and Meaning in American Sign Language. Cambridge ; New York: Cambridge University Press.  
 Lugaresi, C. et al. (2019) “MediaPipe: A Framework for Perceiving and Processing Reality.” In Third Workshop on Computer Vision for AR/VR at IEEE Computer Vision and Pattern Recognition (CVPR) 2019. Available at: https://mixedreality.cs.cornell.edu/s/NewTitle\_May1\_MediaPipe\_CVPR\_CV4ARVR\_Workshop\_2019.pdf (Accessed: May 2, 2023).  
 Luqman, H. and Mahmoud, S.A. (2020) “A Machine Translation System from Arabic Sign Language to Arabic.” Universal Access in the Information Society, 19(4), pp. 891–904. DOI: 10.1007/s10209-019-00695-6.  
 Luvizon, D., Picard, D. and Tabia, H. (2018) “2D/3D Pose Estimation and Action Recognition Using Multitask Deep Learning.” In pp. 5137–5146. DOI: 10.1109/CVPR.2018.00539.  
 Martinez, G.H. (2019) “OpenPose: Whole-Body Pose Estimation.”  
 Martinez, J. et al. (2017) “A Simple Yet Effective Baseline for 3d Human Pose Estimation.” In 2017 IEEE International Conference on Computer Vision (ICCV). 2017 IEEE International Conference on Computer Vision (ICCV). Venice: IEEE, pp. 2659–2668. DOI: 10.1109/ICCV.2017.288.  
 Md Asif Jalal, R.C.R.K.M. a L.M. (2018) “American Sign Language Posture Understanding.” In International Conference on Information Fusion (FUSION. pp. 573–579.  
 Meng, L. and Gao, H. (2021) “3D Human Pose Estimation Based on a Fully Connected Neural Network With Adversarial Learning Prior Knowledge.” Frontiers in Physics, 9. Available at: https://www.frontiersin.org/articles/10.3389/fphy.2021.629288 (Accessed: May 9, 2023).  
 Misra, D. (2020) (arXiv:1908.08681) Available at: http://arxiv.org/abs/1908.08681 (Accessed: September 2, 2023).  
 Montesinos López, O.A., Montesinos López, A. and Crossa, J. (2022) “Fundamentals of Artificial Neural Networks and Deep Learning.” In Multivariate Statistical Machine Learning Methods for Genomic Prediction. Cham: Springer International Publishing, pp. 379–425. DOI: 10.1007/978-3-030-89010-0\_10.  
 Moryossef, A. et al. (2021) “Evaluating the Immediate Applicability of Pose Estimation for Sign Language Recognition.” In 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 3429–3435. DOI: 10.1109/CVPRW53098.2021.00382.  
 NIDCD. (2021) What Is American Sign Language (ASL)? | NIDCD. Available at: https://www.nidcd.nih.gov/health/american-sign-language (Accessed: May 13, 2023).  
 O’Shea, K. and Nash, R. (2015) “An Introduction to Convolutional Neural Networks.” ArXiv. Available at: https://www.semanticscholar.org/paper/An-Introduction-to-Convolutional-Neural-Networks-O%E2%80%99Shea-Nash/f46714d200d69eb9cb5cce176297b89a3f5e3a2c (Accessed: May 11, 2023).  
 Othman, A. and Jemni, M. (2012) “English-ASL Gloss Parallel Corpus 2012: ASLG-PC12.”  
 Pathak, A. and Maheshwari, R. (2019) “Comparative Analysis of Different Loss Functions for Deep Face Recognition.” In Proceedings of the 2019 2nd International Conference on Algorithms, Computing and Artificial Intelligence. ACAI 2019: 2019 2nd International Conference on Algorithms, Computing and Artificial Intelligence. Sanya China: ACM, pp. 390–397. DOI: 10.1145/3377713.3377779.  
 Patterson, J. and Gibson, A. (2017) Deep Learning: A Practitioner’s Approach. first edition. Sebastopol, CA: O’Reilly.  
 Phankokkruad, M. and Wacharawichanant, S. (2019) “A Comparison of Efficiency Improvement for Long Short-Term Memory Model Using Convolutional Operations and Convolutional Neural Network.” In 2019 International Conference on Information and Communications Technology (ICOIACT). 2019 International Conference on Information and Communications Technology (ICOIACT). pp. 608–613. DOI: 10.1109/ICOIACT46704.2019.8938410.  
 Ramachandran, P., Zoph, B. and Le, Q.V. (2018) “Searching for Activation Functions.” Available at: https://openreview.net/forum?id=Hkuq2EkPf (Accessed: September 2, 2023).  
 Rastgoo, R., Kiani, K. and Escalera, S. (2021) “Sign Language Recognition: A Deep Survey.” Expert Systems with Applications, 164, p. 113794. DOI: 10.1016/j.eswa.2020.113794.  
 Redmon, J. et al. (2016) “You Only Look Once: Unified, Real-Time Object Detection.” In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA: IEEE, pp. 779–788. DOI: 10.1109/CVPR.2016.91.  
 Rosero-Montalvo, P.D.G.-T.P.F.-B.E.C.-G.J.O.-P.S.B.-P.H.&. P.-O.D.H. (2018) Sign Language Recognition Based on Intelligent Glove Using Machine Learning Techniques. IEEE Third Ecuador Technical Chapters Meeting (ETCM.  
 Sahoo, A., Mishra, G. and Ravulakollu, K. (2014) “Sign Language Recognition: State of the Art.” ARPN Journal of Engineering and Applied Sciences, 9, pp. 116–134.  
 Salehinejad, H. et al. (2017) “Recent Advances in Recurrent Neural Networks.”  
 Sarma, D. and Bhuyan, M.K. (2021) “Methods, Databases and Recent Advancement of Vision-Based Hand Gesture Recognition for HCI Systems: A Review.” SN Computer Science, 2(6), p. 436. DOI: 10.1007/s42979-021-00827-x.  
 Schmidhuber, J. (2015) “Deep Learning in Neural Networks: An Overview.” Neural Networks, 61, pp. 85–117. DOI: 10.1016/j.neunet.2014.09.003.  
 Selvaraj, P. et al. (2022) “OpenHands: Making Sign Language Recognition Accessible with Pose-Based Pretrained Models across Languages.” In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). ACL 2022. Dublin, Ireland: Association for Computational Linguistics, pp. 2114–2133. DOI: 10.18653/v1/2022.acl-long.150.  
 Sharvani Srivastava, A.G.R.M.S.S. (2021a) “Sign Language Recognition System Using TensorFlow.” In International Conference on Advanced Network Technologies and Intelligent Computing.  
 Sharvani Srivastava, A.G.R.M.S.S. (2021b) “Sign Language Recognition System Using TensorFlow.” In International Conference on Advanced Network Technologies and Intelligent Computing. DOI: 10.1007/978-3-030-96040-7\_48.  
 Singh, A. (2022) Challenges for Sign Language Translation. Medium. Available at: https://blog.ml6.eu/challenges-for-sign-language-translation-5d1062ee91a (Accessed: May 15, 2023).  
 Stokoe, W.C., Jr. (2005) “Sign Language Structure: An Outline of the Visual Communication Systems of the American Deaf.” The Journal of Deaf Studies and Deaf Education, 10(1), pp. 3–37. DOI: 10.1093/deafed/eni001.  
 Subramanian, B. et al. (2022) “An Integrated Mediapipe-Optimized GRU Model for Indian Sign Language Recognition.” Scientific Reports, 12, p. 11964. DOI: 10.1038/s41598-022-15998-7.  
 Sunmok Kim, Y.J. a K.-B.L. (2018) “An Effective Sign Language Learning with Object Detection Based ROI Segmentation.” In 2018 Second IEEE International Conference on Robotic Computing. pp. 330–333.  
 Supalla, S.J., Cripps, J.H. and Byrne, A.P.J. (2017) “Why American Sign Language Gloss Must Matter.” American Annals of the Deaf, 161(5), pp. 540–551. DOI: 10.1353/aad.2017.0004.  
 Terven, J. and Cordova-Esparza, D.-M. (2023) A Comprehensive Review of YOLO: From YOLOv1 to YOLOv8 and Beyond.  
 Thompson, N.C. et al. (2020) “THE COMPUTATIONAL LIMITS OF DEEP LEARNING.” 4.  
 Uzair, M. and Jamil, N. (2020) “Effects of Hidden Layers on the Efficiency of Neural Networks.” In 2020 IEEE 23rd International Multitopic Conference (INMIC). 2020 IEEE 23rd International Multitopic Conference (INMIC). pp. 1–6. DOI: 10.1109/INMIC50486.2020.9318195.  
 Walczak, S. and Cerpa, N. (2003) “Artificial Neural Networks.” In Meyers, R.A. (ed.) Encyclopedia of Physical Science and Technology (Third Edition). New York: Academic Press, pp. 631–645. DOI: 10.1016/B0-12-227410-5/00837-1.  
 Wang, M., Tighe, J. and Modolo, D. (2020) “Combining Detection and Tracking for Human Pose Estimation in Videos.” In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Seattle, WA, USA: IEEE, pp. 11085–11093. DOI: 10.1109/CVPR42600.2020.01110.  
 Wang, Y. et al. (2022) “A Comprehensive Review of Modern Object Segmentation Approaches.” Foundations and Trends® in Computer Graphics and Vision, 13(2–3), pp. 111–283. DOI: 10.1561/0600000097.  
 WHO. (2023) Deafness and Hearing Loss. Available at: https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss (Accessed: May 16, 2023).  
 Wiley, D.J.F. (2016) R Deep Learning Essentials. Packt Publishing.  
 Yamak, P.T. and P. K. G, L.Y. (2019) “A Comparison between ARIMA, LSTM, and GRU for Time Series Forecasting.” In Computing and Artificial Intelligence (ACAI’. Sanya, China.  
 Ying Wu. and Huang, T.S. (1999) “Human Hand Modeling, Analysis and Animation in the Context of HCI.” In Proceedings 1999 International Conference on Image Processing (Cat. 99CH36348). 6th International Conference on Image Processing (ICIP’99). Kobe, Japan: IEEE, pp. 6–10. DOI: 10.1109/ICIP.1999.817058.  
 Yu, Y. et al. (2023) “Techniques and Challenges of Image Segmentation: A Review.” Electronics, 12(5), p. 1199. DOI: 10.3390/electronics12051199.  
 Zeshan, U. (2006) “Sign Language of the World.” In Encyclopedia of Language and Linguistics (Vol. 11). Elsevier, pp. 358–365. DOI: 10.1016/B0-08-044854-2/00243-1.  
 Zhang, X., Zou, Y. and Shi, W. (2017) “Dilated Convolution Neural Network with LeakyReLU for Environmental Sound Classification.” In 2017 22nd International Conference on Digital Signal Processing (DSP). 2017 22nd International Conference on Digital Signal Processing (DSP). pp. 1–5. DOI: 10.1109/ICDSP.2017.8096153.  
 Zhang, Z. (2012) “Microsoft Kinect Sensor and Its Effect.” IEEE MultiMedia, 19(2), pp. 4–10. DOI: 10.1109/MMUL.2012.24.  
 Zhao, Z.-Q. et al. (2019) “Object Detection With Deep Learning: A Review.” IEEE Transactions on Neural Networks and Learning Systems, 30(11), pp. 3212–3232. DOI: 10.1109/TNNLS.2018.2876865.