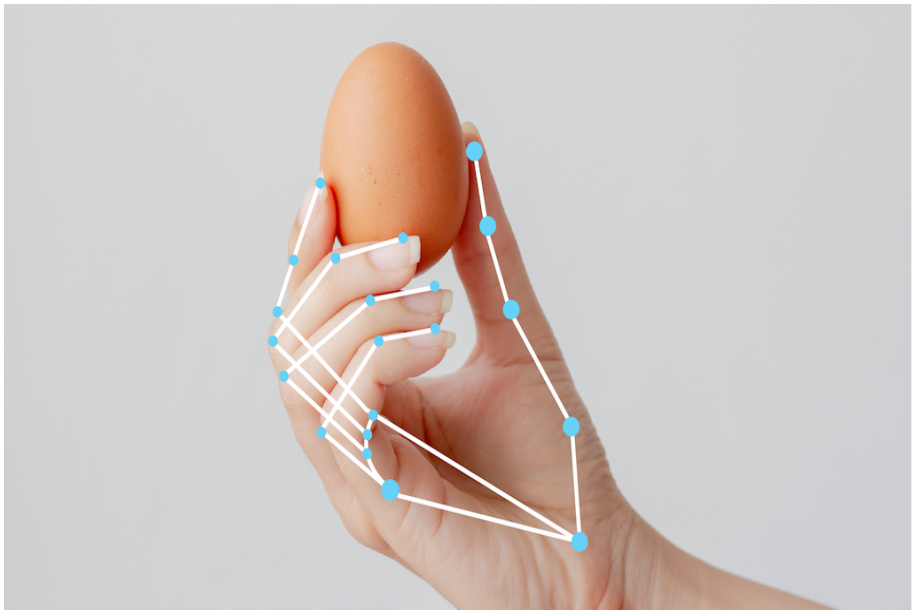

# Utilização de Inteligência Artificial na Tradução de Língua Gestual para Língua Verbal


**TIAGO FILIPE DA SILVA GONÇALVES**

(Licenciado)


Dissertação para obtenção do Grau de Mestre em

Engenharia Informática e Multimédia


**Orientadores:** Doutor Arnaldo Joaquim Castro Abrantes

Doutor Pedro Miguel Torres Mendes Jorge


**Júri:**

**Presidente:** Doutor Carlos Jorge de Sousa Gonçalves

**Vogais:** Doutor Gonçalo Caetano Marques

Doutor Arnaldo Joaquim Castro Abrantes

**Novembro 2024**


# Utilização de Inteligência Artificial na Tradução de Língua Gestual para Língua Verbal

**TIAGO FILIPE DA SILVA GONÇALVES**

(Licenciado)

Dissertação para obtenção do Grau de Mestre em

Engenharia Informática e Multimédia

**Orientadores:** Doutor Arnaldo Joaquim Castro Abrantes, ISEL
Doutor Pedro Miguel Torres Mendes Jorge, ISEL

**Júri:**

**Presidente:** Doutor Carlos Jorge de Sousa Gonçalves, ISEL
**Vogais:** Doutor Gonçalo Caetano Marques, ISEL
Doutor Arnaldo Joaquim Castro Abrantes, ISEL

**Novembro 2024**

*Aos meus avós paternos.*

# Agradecimentos

Agradeço aos meus Orientadores de projeto e Coordenador de Curso por terem aceite esta proposta de Trabalho Final de Mestrado, tal como toda a ajuda e disponibilidade ao longo da realização do projeto.

Agradeço a todos os colegas de turma, amigos e familiares que me ajudaram na criação de uma base de dados de Língua Gestual Portuguesa.

# Resumo

Numa altura em que o ser humano tem a capacidade de comunicar em qualquer língua, com sistemas de tradução disponíveis a um clique de distância, as línguas gestuais permanecem excluídas.

O principal objetivo deste projeto é desenvolver um sistema prático e moderno, que permita a tradução de língua gestual para língua verbal, de forma tão acessível e imediata quanto as traduções entre línguas verbais. Para tal, é proposto a utilização de modelos de Inteligência Artificial que, através da utilização de uma câmara e de uma saída ajustada ao recetor, seja um visor ou um auscultador, captam os gestos em língua gestual e traduzem para língua verbal. Assim, no processo de identificação dos elementos de comunicação, este sistema define-se como o canal, ou seja, o emissor é filmado enquanto comunica a mensagem em língua gestual, e o recetor, através de um visor ou auscultador, recebe a mensagem já traduzida em língua verbal.

Para o desenvolvimento deste sistema, foram treinadas várias redes neuronais utilizando a base de dados SIGNUM, de Língua Gestual Alemã. Além disso, recorreu-se à ferramenta MediaPipe para a tarefa de pré-processamento dos dados, garantindo uma manipulação de dados de tamanho reduzido em comparação com imagens ou vídeos.

Por fim, selecionou-se os modelos que obtiveram os melhores resultados, e desenvolveu-se uma aplicação que traduz língua gestual para língua verbal.

**Palavras-chave:** Língua Gestual, Tradução, SIGNUM, Aprendizagem Automática, Redes Neuronais, Transformer, MediaPipe, Reconhecimento de Língua Gestual Contínuo

# Abstract

In an era where humans can comunicate in any language with a translation systems available at the click of a button, sign languages remain largely excluded.

The main objetive of this project, is to develop a pratical and modern system, that enables the translation of sign language into spoken language, in a manner that is as accessible and instantaneous as current spoken language translation. For that, the proposed solution involves the use of Artificial Intelligence models, which, by utilizing a camera and a display or sound device, capture signs and translate them into words. So, in this communication process, the system functions as a channel, where the sender is recorded while communicating his message in sign language, and the receiver, through a display or sound device, receives a translated message.

To develop this system, several neural networks were trained using the SIGNUM dataset, a German Sign Language dataset. Additionally, the MediaPipe tool was used for data preprocessing tasks, allowing the handling of a data smaller than images or videos.

In the end, the models that achieved the best results were selected, and an application was implemented to translate sign language into spoken language.

**Keywords:** Sign Language, Translation, SIGNUM, Machine Learning, Neural Networks, Transformer, MediaPipe, Continuous Sign Language Recognition

# Índice

# Índice de Figuras

# Índice de Tabelas

# Índice de Listagens

# Glossário

| | |
|---|---|
| Abade | O termo Abade é um título eclesiástico utilizado para se referir ao superior de um mosteiro ou abadia |
| Datilologia | Datilologia é soletrar para falar, em língua gestual. Este conceito será explicado na secção 3.1 |
| Gloss | Termo utilizado para associar uma palavra a um gesto |
| Linguagem Mímica | A Linguagem Mímica, ou Linguagem de "Estampas" (traduzido do francês *estampes*) é uma linguagem sem estrutura, onde é possível representar objetos e ações, e até contar histórias, através de gestos que se aproximam das formas e movimentos dos elementos que se pretende representar |
| Repetidor | Professor que repete as aulas |

# Siglas

| | |
|---|---|
| ASL | Língua Gestual Americana (*American Sign Language*) |
| CNN | Rede Neuronal Convolucional (*Convolutional Neural Network*) |
| DGS | Língua Gestual Alemã (*Deutsche Gebärdensprache*) |
| DTW | Alinhamento Dinâmico de Tempo (*Dynamic Time Warping*) |
| HMM | Modelos de Markov Não-Observáveis (*Hidden Markov Model*) |
| LGP | Língua Gestual Portuguesa |
| Libras | Língua Gestual Brasileira |
| MLP | Rede Neuronal de Perceptrão Multicamada (*Multilayer Perceptron*) |
| NLP | Processamento de Linguagem Natural (*Natural Language Processing*) |
| RNN | Rede Neuronal Recurrente (*Recurrent Neural Network*) |
| TCNN | Rede Neuronal Convolucional Temporal (*Temporal Convolutional Neural Network*) |

# 1 Introdução

A Língua Gestual é o principal meio comunicação da comunidade Surda. A sua estrutura é composta por um conjunto de gestos, expressões faciais e expressões corporais, onde cada uma contém as suas próprias regras gramaticais.

Diferente da língua verbal, o meio mais comum de interação entre indivíduos, e que depende da emissão e receção de sons. A língua gestual distingue-se pela sua natureza visual-motora, sendo a sua emissão realizada através de gestos, e a sua receção através da visão. Esta diferença no canal utilizado para transmitir a mensagem, faz também com que a sua tradução tenha que ser realizada de forma diferente às línguas verbais, onde primeiro é necessário um entendimento e reconhecimento visual.

## 1.1 Motivação

A motivação para a realização deste trabalho não surgiu da noite para o dia, e é um problema que sempre esteve presente na minha vida pessoal.

Há alguns anos, foi-me pedido para desenvolver um projeto final para um curso focado em Dispositivos Móveis. Na época, apesar de não ter quaisquer bases em Inteligência Artificial, dedicava alguma parte do meu tempo livre ao estudo de algoritmos introdutórios nesta área, o que me despertou o interesse de realizar um projeto que tocasse em Dispositivos Móveis e Inteligência Artificial.

A decisão final sobre o tema do projeto foi inspirada por uma situação pessoal durante uma conversa em família. Enquanto discutíamos a infância do meu pai, surgiu a necessidade de confirmar um detalhe com a minha avó paterna, e recordo-me, que devido à minha avó estar presente, mas não ter estado a participar na conversa até então, insinuámos algo como "Vamos ter que repetir para a avó perceber o que estávamos a falar", pois a minha avó paterna é surda. Mas estávamos enganados, pois logo a seguir a minha avó explicou aquilo que precisávamos.

É relevante contextualizar que meus avós paternos são surdos, e a minha avó, além de ler muito bem os lábios, consegue produzir sons percetíveis, o que permite uma comunicação em Língua Portuguesa, embora com algumas dificuldades. Esta mesma ideia de "permite uma comunicação em Língua Portuguesa" foi o que me levou a crer que talvez fosse possível desenvolver uma aplicação que traduz em tempo real a Língua Gestual Portuguesa (LGP), permitindo à minha avó expressar-se na sua língua.

Na altura acabei por desenvolver uma aplicação móvel para iPhone, que utilizava aplicações e bibliotecas da Apple, como, Core ML e Create ML, para traduzir os números da Língua Gestual Portuguesa (gestos estáticos) em tempo real. Mais tarde, durante a licenciatura, apliquei a mesma abordagem, com gestos dinâmicos, adicionando gestos básicos como cores, profissões e meses do ano, e no fim conseguia alternar entre o reconhecimento de gestos estáticos e gestos dinâmicos.

Atualmente, no Mestrado com bastante foco em Inteligência Artificial, decidi, eu próprio, programar e treinar um modelo. Ou seja, a motivação para este trabalho apresenta um grande vertente pessoal, e tem como objetivo dar a uma pessoa surda a possibilidade de se expressar na sua própria língua, sem a necessidade de se adaptar aos restantes.

## 1.2 Objetivo

Para proporcionar a uma pessoa surda a possibilidade de se expressar na sua própria língua, é necessário criar um sistema móvel capaz de traduzir língua gestual para língua verbal. Neste sentido, este trabalho propõe investigar a utilização de modelos de Inteligência Artificial para a tradução de língua gestual. Os modelos explorados não deveram ser computacionalmente exigentes, pois devem ser acessíveis para todas as plataformas .

## 1.3 Contribuições

As contribuições do trabalho apresentado neste documento são as seguinte:

- Protótipo de uma aplicação que traduz em tempo real língua gestual para língua verbal;
- Utilização e análise da ferramenta MediaPipe, através de vários modelos, para o problema de tradução de Língua Gestual;
- Adaptação da arquitetura Transformer para o reconhecimento de Língua Gestual Contínua;
- Extensão do vocabulário da base de dados SIGNUM, através de gestos isolados.

## 1.4 Organização do Documento

O documento é composto por sete capítulos, cada um importante para a contextualização do projeto.

No Capítulo 1, **Introdução**, ou capítulo atual, demonstra-se o problema que este projeto pretende solucionar, como também, a motivação, os objetivos e a estrutura do projeto.

No capítulo 2, **Língua Gestual**, é apresentado um breve resumo da história da língua gestual e Língua Gestual Portuguesa, termos utilizados para o estudo da língua gestual e uma contextualização dos tipos de reconhecimento da língua gestual.

No capítulo 3, **Estado da Arte**, é introduzido alguns trabalhos relacionados com o reconhecimento visual, reconhecimento de gestos, reconhecimento de língua gestual e outros trabalhos que apresentam métodos diferentes para a tradução de língua gestual.

No capítulo 4, **Modelo Proposto**, é partilhado a escolha de entre as várias ferramentas e técnicas possíveis para a realização deste problema. De seguida, é detalhada as bases de dados escolhidas, e explicadas a ferramenta MediaPipe.

No capítulo 5, **Classificação**, é apresentado os vários modelos de classificação utilizados e os seus resultados, tal como as alterações necessárias para a aplicação de *landmarks*.

No capítulo 6, **Implementação**, é descrita a implementação de uma aplicação que traduz em tempo real língua gestual para língua verbal, através dos modelos Transformer anteriormente treinados, e outras ferramentas.

No capítulo 7, **Conclusão**, é feita uma breve conclusão sobre o trabalho desenvolvido e possíveis trabalhos futuros.

# 2 Língua Gestual

> "*So long as there are two deaf people upon the face of the earth and they get together, so long will signs be in use.*"
>
> J. Schuyler Long, *The Sign Language: A Manual of Signs*

Este capítulo encontra-se dividido em quatro secções. Na primeira secção, é apresentado um resumo da história da língua gestual, com principal foco na educação. Na segunda secção, são partilhados os parâmetros que definem o estudo da língua gestual. Na terceira secção, é retomada a história da língua gestual, mas desta vez em Portugal. Na quarta secção, introduz-se os termos utilizados no reconhecimento de língua gestual.

## 2.1 História da Língua Gestual

Uma "língua natural", de acordo com a definição do Dicionário Online Priberam da Língua Portuguesa [1], é um "sistema linguístico que é língua materna de algum grupo humano e é usado naturalmente como meio de comunicação por indivíduos que a aprenderam, por oposição a língua artificial". Dada esta definição, pode-se afirmar que uma língua gestual é uma língua natural, uma vez que, através de uma vertente visual-gestual ou visual-motora, serve como meio de comunicação para muitos indivíduos Surdos e/ou com dificuldades auditivas, que a aprendem e a utilizam naturalmente como sua língua materna. No entanto, as línguas gestuais nem sempre foram reconhecidas desta forma.

Durante séculos, as línguas gestuais não foram reconhecidas como línguas completas. As mesmas eram frequentemente percebidas como um sistema de gestos, sem a estrutura gramatical complexa das línguas verbais. Esta perceção, mais os preconceitos sociais e supostas conclusões como a de que os Surdos não tinham memória, nem compreendiam ideias abstratas [2], levou as línguas gestuais ao seu subdesenvolvimento, impactando a vida das comunidades Surdas. A visão predominante considerava a surdez como uma deficiência a ser superada, em vez de uma diferença a ser acolhida. Esta perspetiva influenciou

significativamente a educação, focando em métodos orais, e tendo como objetivo ensinar o Surdo a falar e a ler lábios, em vez de aprender e desenvolver uma língua gestual [3].

Contudo, uma mudança surgiu com o Abade Charles-Michel de l'Épée, um padre francês do século XVIII que desempenhou um papel significativo na língua gestual. L'Épée deparou-se com a situação de duas crianças gémeas, anteriormente instruídas a comunicar-se através de uma Linguagem Mímica pelo seu recém falecido tutor, Padre Vanin. Sensibilizado, decidiu acolhê-las e educá-las, o que levou-o a investigar e criar o Método de Sinais, e mais tarde a estabelecer a primeira escola pública para Surdos em Paris [4].

No entanto, esta abordagem foi influenciada pela língua francesa e embora não se tratasse de uma Datilologia propriamente dita, era notoriamente semelhante. Por exemplo, para expressar a palavra "dá" (da frase "ele dá farofa aos pássaros"), eram necessários cinco gestos distintos: um para o verbo, outro para o tempo presente, outro para a terceira pessoa, outro para o singular, e finalmente o gesto específico de "dar" [5]. O sistema ainda apresentava influências da gramática francesa e a sua "Datilologia baseada na gramática" não era nem simples, nem intuitiva. Apesar do seu objetivo de facilitar a comunicação e a educação da comunidade Surda, este método acabou por reforçar a perceção da língua gestual como uma derivação da língua verbal.

Embora bem intencionada, este ênfase no oralismo e a utilização da língua gestual principalmente como uma ferramenta para a aprendizagem ou aproximação à língua verbal dominaram a educação durante décadas. Contudo, o mais significativo foi o facto de se ter iniciado este processo, já que o sucessor de l'Épée, o Abade Sicard, acabaria por promover sessões públicas assistidas por monarcas, filósofos e educadores. Estes, ao regressarem aos seus países, fundaram Institutos para Surdos inspirados no modelo francês [6]. Um exemplo notável é o dos Estados Unidos, onde Laurent Clerc, educador do Instituto Francês de Surdos, e Thomas Gallaudet, um educador americano, após assistirem a algumas destas sessões em Paris, viajaram juntos para a América e fundaram a Escola Americana para Surdos (*American School for the Deaf*) [3, 7].

Mesmo com o ensino da língua gestual ainda longe do ideal, ao longo dos anos foi progredindo, especialmente porque Surdos como Jean Massieu começaram a lecionar, e a apresentar uma perspetiva que os ouvintes não possuíam [5, 6]. Este desenvolvimento formou uma comunidade, o que, possivelmente, foi o fator mais importante de todos. Vários estudos e livros apontam que a Língua Gestual Francesa não foi criada depois da fundação da primeira escola, mas sim antes, e que, fora do ambiente escolar, os alunos utilizavam uma língua gestual distinta daquela ensinada em aula, algo que viria a repetir noutros países [5, 8]. As formações destas comunidades garantiu que as línguas gestuais não caíssem em desuso, permitindo a evolução de cada uma e, o mais importante, a sua utilização e aprendizagem de forma natural.

O Método de Sinais, como outros métodos, foi progressivamente caindo em desuso com o passar dos anos [5]. A educação foi evoluindo e as aulas começaram a focar-se na língua que os alunos falavam fora das aulas. Embora no século XIX, em alguns países, a educação dos Surdos fosse já focada na língua que uma comunidade tinha criado e evoluído ao longo de anos, a mudança crucial na compreensão e reconhecimento da língua gestual só ocorreu em meados do século XX, com o trabalho de William Stokoe, um linguista da Universidade Gallaudet [9].

A publicação de Stokoe em 1960, "Sign Language Structure: An Outline of the Visual Communication Systems of the American Deaf" [9], desafiou as suposições predominantes, ao estudar pela primeira vez a língua gestual como um sistema linguístico. Ele identificou um conjunto de parâmetros: configuração da mão, localização e movimento, a que deu o nome de queremas (*cheremes*), equivalentes aos fonemas das línguas verbais, que ao se juntarem obtinha-se o gesto. Provando de vez que a língua gestual não é uma mera representação visual das línguas verbais.

## 2.2 Estudo da Língua Gestual

Os termos propostos por William Stokoe como querologia (*cherology*) e queremas (*cheremes*) não prevaleceram, e a terminologia correta acabou por ser "fonologia" e "fonemas" e outros termos idênticos aos utilizados no estudo normal de uma língua. Ainda assim os parâmetros identificados por Stokoe permaneceram, e apenas foram acrescentados mais [9–12], que incluem:

- **Configuração das mãos** - Forma que as mãos assumem durante a execução de um sinal. Cada língua gestual tem um conjunto específico de configurações possíveis, e estas configurações são fundamentais para diferenciar sinais que de outra forma seriam idênticos em localização e movimento.
- **Localização** - Ponto no espaço onde o sinal é realizado. A mudança na localização pode alterar completamente o significado de um sinal, tal como uma alteração na entoação pode mudar o significado de uma palavra falada.
- **Movimento** - Um dos elementos mais dinâmicos das línguas gestuais, que envolve deslocamentos lineares, curvilíneos, ou até mesmo vibrações e batimentos. O movimento contribui significativamente para o significado do sinal, e apresenta uma relação temporal, e caso seja ignorada, ao se captar um só instante de todo o movimento, o sinal perde todo o seu significado.
- **Orientação das mãos** - Refere-se à direção em que a palma e dedos estão voltados durante a execução de um sinal. Uma mudança na orientação pode modificar o significado de um sinal, da mesma forma que um som em uma palavra falada.

- **Postura corporal e expressões faciais** - Também tratado por "gestos não manuais" são todos os movimentos que não estão ligados à mão, como cabeça, lábios, bochechas, sobrancelhas, ombros e entre outros. Normalmente indicam perguntas, exclamações, negações ou intensificações, funcionando de maneira semelhante à entoação nas línguas verbais.

Outros parâmetros foram propostos, como por exemplo, **condição de simetria** (*symmetry condition*), que explora a orientação de ambas as mãos, de forma idêntica e oposta, ou por exemplo, a **condição de dominância** (*dominance condition*), que explora a existência de movimento por parte da mão dominante e uma configuração estática por parte da mão não-dominante [12].

Pode-se dizer que é neste ponto que os estudos divergem. Porque por exemplo, a Língua Gestual Indo-Paquistanês apresenta uma forte componente "não manual", onde gestos como *BAHUT_ACHA* "excelente" e *VAHI:* "mesmo" se distinguem pelo fechar da boca [13]. Cada língua gestual é um caso, e a necessidade de dar mais ou menos relevância a certos parâmetros varia consoante cada uma, seja por influências de outras línguas ou por motivos geográficos/culturais.

## 2.3 Língua Gestual Portuguesa

O envolvimento de Portugal na educação de Surdos teve origem, no século XVIII, com o português Jacob Rodrigues Pereira, que desempenhou seu papel fora do país. No livro, escrito pelo Abade Charles-Michel de l'Épée [4], é mencionado que, antes do próprio Abade, outros já se dedicavam à educação de Surdos. Entre eles Pereira, que se destacou pelo seu trabalho e pela formação do discípulo Saboureux de Fontenay.

Em Portugal, os avanços na educação de surdos só começaram no século XIX, com a fundação do Real Instituto dos Surdos-Mudos e Cegos em 1823, a pedido da princesa D. Isabel Maria, filha do Rei D. João VI [8, 14]. Este instituto foi estabelecido por Pär Aron Borg, também fundador do Instituto Público de Cegos e Surdos em Manilla, na Suécia [15], e tinha como principal objetivo educar surdos e cegos, integrando-os na sociedade.

Em 1828 Pär Aron Borg voltou para a Suécia, e o seu irmão, Joham Borg, assumiu a direção do instituto até seu falecimento em 1833 [16]. Após a morte de Joham Borg, José Crispim da Cunha, que anteriormente exercia o cargo de repetidor e terceiro professor do instituto, assumiu a direção. Um ano mais tarde, ocorreu a fusão do instituto com a Casa Pia de Lisboa [16].

A transição da direção e fusão com a Casa Pia envolveu motivos políticos, relacionados com a Guerra Civil Portuguesa (1832-1834), como também questões financeiras, popularidade do instituto e queixas/criticas dos alunos mais velhos comparando Pär Aron Borg

a Joham Borg e José Crispim da Cunha, que não serão exploradas neste projeto. Mas é importante salientar o seguinte: Era utilizado um ensino ainda baseado na oralidade [17], Pär Aron Borg já enfrentava conflitos com a Casa Pia, deixou a direção e foi sucedido por dois diretores, um dos quais faleceu e o seguinte esteve no cargo cerca de um ano, até se demitir durante o processo de fusão. Como resultado, os alunos do instituto foram temporariamente colocados sob a responsabilidade do "melhor" aluno, enquanto se procurava um novo professor [16], algo que nunca aconteceu. O ensino de surdos na Casa Pia de Lisboa foi oficialmente extinto em 1860 [18].

Após o encerramento do Real Instituto dos Surdos-Mudos e Cegos e regularização do ensino pelo Decreto de 17 de Junho de 1870 [19], o ensino de Surdos em Portugal tomou um novo rumo. Várias instituições foram fundadas ou abriram portas para a educação de Surdos [17], destacando o Instituto de Surdos-Mudos, em Guimarães, fundado em 1870, por Pedro Maria de Aguilar, que segundo D.António da Costa [8]:

> "N'este ponto ha uma novidade curiosa. Nunca lhes foram impostos signaes do alphabeto pelos dedos, systema ainda hoje na Europa geralmente usado. Não é o professor que decreta a linguagem mimica, mas os próprios mudos é que estabeleceram os signaes da conversação, conforme a própria rasão lh'os indicava. Instituiram a sua linguagem, natural, espontânea, e os mestres foram-na recebendo, desprezando as theorias dos signaes methodicos, pouco racionaes."

Podendo concluir que no instituto de Pedro Maria de Aguilar, a intuição dos alunos era mais importante que os métodos anteriormente criados.

O ensino de Surdos até 1894, manteve-se no domínio particular, mas, embora as boas intenções dos seus fundadores, um instituto ou atividade financiada por quem a criou, não só era temporária como também era de alguma forma única. Isto é, muitos destes projetos duraram pouco mais que uma década [17], e cada uma semeava o conhecimento da forma que conhecia, criando barreiras sociais numa comunidade só. Foi então que em 1894, pelo Decreto de 20 de Novembro de 1894, que é aprovada o ensino público de Cegos e Surdos [20].

Não deverá ter sido um início fácil, na altura, várias deveriam ser as línguas gestuais faladas em Portugal (por exemplo, os meus avôs paternos, ambos residentes em Lisboa, aprenderam em institutos diferentes, e as suas línguas gestuais apresentavam diferenças). Mas a existência de um ensino contínuo levou à criação do que conhecemos hoje, como Língua Gestual Portuguesa, que em 1997 foi reconhecida como língua oficial de Portugal, junto do Português e Mirandês [21].

# 3 Estado da Arte

> "*The road goes ever on and on, down from the door where it began. Now far ahead the road has gone, and I must follow if I can.*"
>
> J.R.R. Tolkien, *The Fellowship of the Ring*

Este capítulo desenvolve sobre os termos utilizados em reconhecimento de gestos e de língua gestual, e apresenta o trabalho realizado em outros projetos. Na primeira secção, são introduzidos termos presentes no reconhecimento de gestos e língua gestual. Na segunda secção, é apresentada a metodologia presente nestes trabalhos, ao mesmo tempo que é realizada a comparação com este projeto. Na terceira secção, são introduzidos dois projetos que saem do âmbito deste trabalho, mas que contribuem para a comunidade Surda.

## 3.1 Reconhecimento de Língua Gestual

O reconhecimento e a segmentação de imagens ou vídeos têm sido uma das investigações com mais destaque na área das Redes Neuronais, e as suas técnicas têm sido utilizadas em múltiplos domínios. Este campo está em constante evolução, desde o desenvolvimento da AlexNet [22] em 2012, marco de uma viragem na utilização de Redes Neuronais para classificação de imagens, ao recém chegado modelo SAM 2 [23] em Julho de 2024, que apresentou um grande avanço na segmentação de imagens.

O objetivo principal é identificar e/ou destacar um ou mais objetos numa imagem ou vídeo, de forma a categorizar corretamente os elementos presentes. No contexto do reconhecimento gestual, um campo do reconhecimento e segmentação de imagens ou vídeos, o foco encontra-se sobre elementos específicos: mãos, face e corpo, e o objetivo é conseguir interpretar as múltiplas possibilidades de configurações, movimentos e localizações que as mãos, face e corpo possam assumir.

Dentro do reconhecimento de gesto, existe ainda o reconhecimento de língua gestual, que concentra-se em aspetos semelhantes ao reconhecimento de gestos. Contudo, acrescenta

uma finalidade ainda mais especifica, porque, para além do reconhecimento e interpretação dos elementos, mãos, face e corpo, este necessita de os traduzir de língua gestual para língua verbal.

Assim como em qualquer outra língua, é possível traduzir a língua gestual. No entanto, ao contrário das línguas verbais, a língua gestual é uma forma de comunicação visual, onde as palavras são substituídas por gestos, levando o seu processamento a ser diferente, e com base em imagens ou vídeos, em vez de áudio.

Ao longo dos anos, vários são os projetos que exploram o tema da tradução de língua gestual [24–35], ainda assim é um problema que continua por ser resolvido. Desde o uso de Correspondência de Modelos (*Template Matching*) [25] a Redes Neuronais Profundas [27–33], diversos são os algoritmos e técnicas explorados, e por isso, neste contexto é importante estabelecer uma distinção clara entre os vários objetivos e focos de cada projeto, começando pela distinção de dois tipo de gestos:

Os **gestos estáticos** [27, 33, 36], são caracterizados pela ausência de movimento. A informação é transmitida através da configuração, localização e orientação específica que a mão assume, sem que haja alteração desses parâmetros durante a execução do gesto. Em outras palavras, o gesto é mantido numa posição fixa por um determinado período de tempo, sendo possível captar o mesmo através de uma única imagem. Exemplos comuns de gestos estáticos na Língua Gestual Portuguesa incluem o alfabeto (na sua maioria, à exceção das letras D, K, Q, W, Y e Z, embora somente a letra Z seja difícil de interpretar através de um único instante/imagem), e os números.

Os **gestos dinâmicos** [25, 26, 28–32, 34, 37], são definidos pelo movimento. Estes envolvem uma transição ou uma sequência de gestos, e a informação é transmitida não só pela configuração e orientação das mãos, mas também pela trajetória do movimento, velocidade, localização e a interação entre as duas mãos e/ou outras partes do corpo. Os gestos dinâmicos podem representar ações, processos ou descrições e são frequentemente mais expressivos e complexos do que os gestos estáticos, sendo possível captar o mesmo apenas com a utilização de um vídeo. Por exemplo, os gestos utilizados para representar profissões, são gestos dinâmicos, e alguns chegam a ilustrar um ato ou ação da própria profissão, como por exemplo, na palavra Juiz, e o ato de proferir uma sentença com o malhete/martelo.

É possível também caracterizar cada projeto através do tipo de tradução que o mesmo implementa, e embora possam existir mais tipos, estes são os mais comuns:

**Datilologia**, **Reconhecimento Datilológico** ou ***Fingerspelling Recognition*** [33], corresponde ao reconhecimento do ato de soletrar palavras através de gestos, onde cada gesto representa uma letra do alfabeto. Este método é frequentemente utilizado para nomes

próprios, termos técnicos, palavras para as quais não existe um gesto específico, ou para quando deseja enfatizar-se uma palavra em particular. Embora seja uma forma viável de tradução de língua gestual, o Reconhecimento Datilológico está longe de ser o método ideal para a tradução de língua gestual, pois não representa a forma natural de se comunicar em língua gestual. Seria como criar uma aplicação *Speech-to-Text*, onde somente é possível reconhecer uma palavra, se a mesma for soletrada.

A dactilologia é um método complementar, de natureza mais lenta e menos fluída em comparação à utilização de gestos específicos para palavras ou conceitos.

**Reconhecimento de Gestos Isolados** ou ***Isolated Gesture Recognition*** [25–27, 29, 30, 32, 36, 37], semelhante ao Reconhecimento Datilológico, este tipo de reconhecimento também se concentra na tradução individual, mas lida com gestos que representam palavras ou conceitos dentro da língua gestual. O Reconhecimento de Gestos Isolados foca-se na tradução de um só gesto, e embora seja possível obter bons resultados com esta abordagem, a mesma destaca-se quando o objetivo é traduzir somente um único gesto, quase como um dicionário ou apoio linguístico.

**Reconhecimento Contínuo de Língua Gestual** ou ***Continuous Sign Language Recognition*** [28, 31], refere-se à tradução da língua gestual na sua forma natural e fluída, abrangendo a sequência completa de gestos e expressões utilizadas em uma conversa. Este método procura implementar o desafio do Reconhecimento de Gestos Isolados ao mesmo tempo que tenta ligar cada um dos gestos. Sendo assim, a interpretação de uma série contínua de gestos, considera não só os gestos individuais mas também a sua concatenação e contexto geral. Ou seja, o reconhecimento contínuo é o método que tenta ”entender” ao máximo a língua gestual, onde é possível traduzir frases inteiras de gestos, em vez de gestos isolados.

Uma das diferenças, mais evidentes, entre o Reconhecimento de Gestos Isolados e Reconhecimento Contínuo de Gestos, está na forma como processam os dados, enquanto o reconhecimento isolado apenas tenta traduzir um único gesto, e após a sua tradução **esquece** o gesto, o reconhecimento contínuo mantêm a **memória** do gesto ajudando não só ao reconhecimento do próximo gesto como também à sua tradução contextual e gramatical.

No presente projeto, a implementação final terá como objetivo a implementação do **Reconhecimento Contínuo de Gestos**. Esta abordagem apresenta semelhanças com os artigos [28, 31], mas irá trabalhar com uma base de dados diferente.

## 3.2 Metodologia

Na implementação e estrutura metodológica, não só dos projetos de reconhecimento de gestos, mas também de reconhecimento visual em geral, é possível identificar a divisão da sua implementação em quatro etapas:

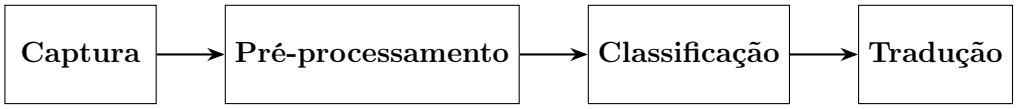


Figura 3.1: *Representação das etapas na implementação de um projeto em reconhecimento visual.*

A primeira etapa, a **Captura**, consiste na captura dos gestos realizados pelo emissor, através de câmaras de vídeo, câmaras de profundidade [29, 30, 37], *webcams* ou de dispositivos móveis. Este processo exige o enquadramento do emissor e recolha dos gestos, expressões faciais e corporais, sendo possível três tipos de captura:

- **Imagem** [27], onde a captura é realizada através de uma única imagem, que tal como já foi mencionado só é útil na interpretação de gestos estáticos;
- **Vídeo** [25, 28–32, 34], onde a mensagem do emissor é primeiro capturada na totalidade, e somente depois é processada e apresentada a tradução;
- **Tempo Real** [26, 33, 36, 37], onde existe uma comunicação constante entre cada etapa, permitindo a tradução no momento.

Cada uma destas capturas apresenta vantagens e desvantagens, ligadas ao método de captura, e capacidade de representar a comunicação gestual de uma forma fidedigna. Tendo sido optado neste projeto, implementar a captura do tipo **Tempo Real**, mas diferentes dos artigos [26, 33, 36, 37]. Pois será realizada um Reconhecimento Contínuo, que tem em conta os gestos anteriormente traduzidos e o contexto da frase.

A segunda etapa, o **Pré-processamento**, é onde são preparadas as imagens ou vídeos para posterior processamento pela etapa da Classificação. O pré-processamento é responsável por ajustar imagens, reduzir o ruído, realçar características ou elementos, e extrair informação, que seja útil para a etapa de Classificação, de onde se destacam as seguintes abordagens:

- O realce ou detenção da cor da pele [25], pode facilitar análises posteriores como, a segmentação das mãos e outras partes do corpo envolvidas na produção de um gesto;
- A aplicação de máscaras ou segmentação das áreas de interesse [27], de forma a isolar, por exemplo, o emissor do fundo da imagem;
- O *crop* da imagem [29], que consiste em cortar a imagem enquadrando apenas as partes relevantes, como as mãos ou face;

- O uso de ferramentas como o MediaPipe [26, 33, 34, 36] ou Apple Vision API [32], que extraem pontos de referência específicos do corpo.

A terceira etapa, a **Classificação**, é onde é realizada a passagem de língua gestual para texto. Atualmente, esta tarefa consiste maioritariamente na utilização de um modelo de Inteligência Artificial como, Rede Neuronal de Perceptrão Multicamada (MLP - *Multilayer Perceptron*) [33], Rede Neuronal Convolucional (CNN - *Convolutional Neural Network*) [27, 29, 30], Rede Neuronal Convolucional Temporal (TCNN - *Temporal Convolutional Neural Network*) e Rede Neuronal Recurrente (RNN - *Recurrent Neural Network*) [28], ou Transformers [31, 32, 34]. Ainda assim é possível utilizar outros algoritmos ou técnicas como Alinhamento Temporal Dinâmico (DTW - *Dynamic Time Warping*) [26], Modelos de Markov Não-Observáveis (HMM - *Hidden Markov Model*) [25, 30, 37], entre outros.

Este projeto irá utilizar a ferramenta MediaPipe [38], e implementar a arquitetura Transformers [39]. Como estabelecido anteriormente, o tipo de reconhecimento deste projeto, é o Reconhecimento Contínuo de Língua Gestual. Será aplicada uma abordagem diferente dos artigos [32, 34], que embora trabalhem com a ferramenta MediaPipe, apresentam um Reconhecimento Isolado, e por isso, tratam os gestos e o tempo (sequências de imagens - *frames*) de forma diferente à entrada do modelo. Também será diferente do artigo [31], pois este aplica o modelo Transformers, mas para imagens RGB, em vez das características (*landmarks*) extraídas da ferramenta MediaPipe.

Destaca-se o trabalho do autor da SIGNUM Dataset [40], que partilha resultados através de *Template Matching* e do Algoritmo de Viterbi (*Viterbi Search*) [41] ou Eigenvoices, Regressão Linear de Máxima Verosimilhança (MLLR - *Maximum Likelihood Linear Regression*), Máximo a posteriori (MAP - *Maximum a Posteriori*) e *Viterbi Search* [42].

A última etapa, a **Tradução**, refere-se à mensagem que o recetor recebe. A forma como a tradução é apresentada varia de acordo com o método de captura escolhido na primeira etapa, Captura. Na captura de uma única imagem, ou vídeo pré-gravado, a mensagem é apresentada de uma só vez. Na captura em tempo real, e presente neste projeto, a mensagem é gradualmente exibida, e no fim, através de um contexto mais claro e de uma avaliação completa da frase, é estabelecido a mensagem final.

## 3.3 Outros Desenvolvimentos

Por fim existem projetos que, apesar de não se enquadrarem diretamente no conceito deste trabalho, apresentam uma abordagem valiosa para a comunidade Surda. Estes projetos, embora diferentes em termos de tecnologia ou método utilizado, mostram que é possível traçar outro caminho, e que nem todas as soluções têm que partir da captura de uma imagem ou vídeo.

O projeto SignAloud [35], desenvolvido na Universidade de Washington, utiliza um par de luvas equipadas com sensores que captam os movimentos e posição das mãos durante a execução dos gestos. Os dados são posteriormente processados, permitindo a tradução dos gestos realizados. Este projeto, apresente um reconhecimento de língua gestual para língua verbal, e demonstra que não é necessário uma câmara, para conseguir-se traduzir língua gestual, e que por isso existem várias formas de abordar este problema.

A aplicação Hand Talk, desenvolvida no Brasil em 2012 [43], implementou uma abordagem inversa ao que se propõe neste projeto. Ao invés de traduzir-se a língua gestual para língua verbal, realiza-se a tradução de língua verbal para língua gestual. Especificamente, esta aplicação permite traduzir um texto ou sinal de fala em Português ou Inglês para Libras (Língua Gestual Brasileira) ou Língua Gestual Americana (ASL - *American Sign Language*). Neste caso é utilizado um *avatar* 3D que executa os gestos, tornando toda a tradução uma animação 3D.

# 4 Modelo Proposto

> "*A film is made three times: first on the page, then on set, and finally in the edit.*"
>
> Cineasta francês, Robert Bresson

O capítulo presente explica a estruturação tomada para a realização do trabalho. Na primeira secção, é apresentado o método proposto, que servirá de guia para este trabalho. Na segunda secção, são explicadas as base de dados utilizadas ao longo do projeto. Na terceira secção, é explicada a ferramenta MediaPipe, e o seu efeito nas bases de dados.

## 4.1 Método Proposto

O objetivo final deste projeto consiste em desenvolver uma tradução de língua gestual para língua verbal, e através da Figura 3.1, é possível preencher e aprofundar cada uma das etapas, que compõem o processo metodológico implementado neste trabalho.

Na primeira etapa, **Captura**, utilizou-se três base de dados. Duas destas base de dados, uma de imagens e outra de vídeos, são base de dados pessoais, previamente adquiridas, e que se focam na Língua Gestual Portuguesa. A outra base de dados, SIGNUM [40], é uma base de dados de Língua Gestual Alemã (DGS - *Deutsche Gebärdensprache*), composta por um total de quatrocentas e cinquenta palavras isoladas e setecentas e oitenta frases, que integram as palavras anteriores, todas elas em formato de vídeo.

Na segunda etapa, **Pré-processamento**, foi escolhida a ferramenta MediaPipe, desenvolvida pela Google Research [38]. Esta ferramenta é *open source*, permite ser executada em tempo real, e foi utilizada com sucesso em outros trabalhos de reconhecimento visual [33, 36]. A ferramenta permite converter vídeo em sequências temporais de *landmarks*, e será explicada na secção 4.3.

A terceira etapa, **Classificação**, é onde se irá treinar modelos de Aprendizagem Profunda,

através dos dados previamente processados. O objetivo desta etapa será identificar de entre os modelos, MLP, CNN e Transformers, qual apresenta melhores resultados. Ainda assim, foi realizada uma implementação preliminar que utiliza o algoritmo DTW, uma técnica clássica, utilizada quando se envolvem sequências temporais.

Por fim, a quarta etapa, é a etapa mais volátil, pois depende das implementações realizadas anteriormente, tal como o tipo de reconhecimento escolhido - Datilológico, Isolado e Contínuo. Sendo então possível definir as quatro etapas da seguinte forma:

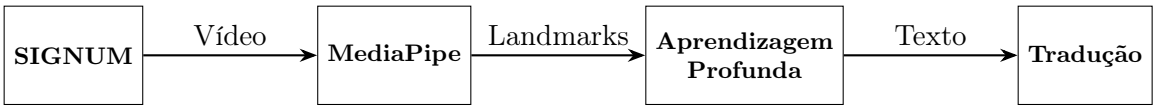


Figura 4.1: *Representação das etapas e fluxo do método proposto para este projeto*

## 4.2 Base de Dados

Durante a implementação deste trabalho utilizou-se três base de dados distintas. As duas primeira são bases de dados pessoais, recolhidas durante a realização de dois projetos anteriores, que a partir deste ponto seram referidas como "LGP-C", onde **C** é o número de classes presente na base de dados. A terceira base de dados utilizada é a SIGNUM Dataset [40], desenvolvida por Ulrich von Agris, cujo o objetivo é a representação de gestos de forma independente do emissor.

### 4.2.1 Base de Dados LGP-9

A base de dados LGP-9 é composta por um conjunto de imagens que contêm os gestos dos números de 1 a 9 em Língua Gestual Portuguesa. Cada imagem representa um só número, e não foi utilizado um fundo constante, nem existe uma distância padrão entre o emissor e a câmara. Ainda assim, todas as imagens focam a mão dominante da pessoa, e por vezes, partes do corpo ou face da pessoa são cortadas da imagem, como é possível observar na Figura 4.3:

***(a)*** ***(b)***

Figura 4.2: *Exemplo de duas imagens da LGP-9.Tradução:* ***(a)*** *Um,* ***(b)*** *Dois.*

Para a obtenção de mais exemplos, foi pedido aos emissores para executar os gestos com ambas as mãos, esquerda e direita, em imagens distintas. Resultando na seguinte distribuição: 60,58% (1479 imagens) correspondentes a gestos feitos com a mão esquerda, e 39,42% (962 imagens) correspondentes a gestos feitos com a mão direita. Mais detalhes, sobre a base de dados, podem ser observados na seguinte Tabela:

Tabela 4.1: Informação das especificações da base de dados LGP-9

| | |
|---|---|
| Nome | LGP-9 |
| Autor | Tiago Gonçalves |
| Língua | Língua Gestual Portuguesa |
| Número de Emissores[a] | 28 |
| Tipo | Imagem |
| Tamanho de Vocabulário | 9 |
| Número de Gestos Estáticos | 9 |
| Número de Gestos Isolados | 9 |
| Número Total de Imagens[b] | 2441 |
| Resolução de Imagem[c] | $3024 \times 4032$ |
| Profundidade de Cor | 24 bpp, RGB |
| Formato da Imagem | JPG |
| Espaço de Armazenamento | 4,33 GB |
| Câmara[c] | iPhone XR |

a) No campo **Número de Emissores**, ou seja, número de pessoas que executam os gestos na base de dados é indicado como sendo 28. Ainda assim, a distribuição dos gestos não é idêntica entre todos, aproximadamente 84% das imagens foram capturadas por apenas três

pessoas, e 16% das imagens pelas restantes 25 pessoas (ou seja, cerca de 44 imagens por classe, dando 2 imagens por pessoa, pois existem pessoas ausentes em certas classes).

b) No campo **Número Total de Imagens** a distribuição de imagens por classe não é uniforme (Respetivamente da classe 1 a 9, a distribuição é a seguinte: 237, 249, 248, 279, 260, 302, 268, 267, 331).

c) Nos campos **Resolução de Imagem** e **Câmara** cerca de 10 imagens por classe, são capturadas por uma câmara desconhecida, e por isso apresentam outra resolução que varia entre $900 \times 1600$ e $1200 \times 1600$.

### 4.2.2 Base de Dados LGP-34

A base de dados LGP-34 é composta por um conjunto de vídeos, todos eles filmados no mesmo ambiente, possuindo um fundo semelhante, e procurando reproduzir um plano americano (enquadramento dos joelhos para cima).

Esta base de dados é composta por gestos dinâmicos, que abrangem três temas, Meses, Cores e Profissões. Além disso, incluí também gestos para uma fruta, uma flor, um país e uma classe nula, todos eles em Língua Gestual Portuguesa.

As classes presentes nesta base de dados são:

- **Meses**: Janeiro, Fevereiro, Março, Abril, Maio, Junho, Julho, Agosto, Setembro, Outubro, Novembro, Dezembro;
- **Cores**: Amarelo, Azul, Branco, Castanho, Cinzento, Cor Laranja, Cor Rosa, Dourado, Lilás, Preto, Verde, Vermelho;
- **Profissões**: Bombeiro, Enfermeiro, Juiz, Médico, Polícia, Professor;
- **Fruta**: Laranja;
- **Flor**: Rosa;
- **País**: Portugal;
- **Nulo**: Posição de Repouso.

As classes ”Cor Laranja” e ”Cor Rosa”, são constituídas por dois gestos dinâmicos. Nestas classes, o gesto ”Cor” é seguido do gesto ”Laranja” ou ”Rosa”, permitindo distinguir os gestos utilizados para a fruta ”Laranja” ou para a flor ”Rosa”. A última classe ”Posição de Repouso”, simboliza a ausência de qualquer gesto, onde o emissor mantém as mãos ao longo do corpo.

Tabela 4.2: Informação das especificações da base de dados LGP-34

| | |
|---|---|
| Nome | LGP-34 |
| Autor | Tiago Gonçalves |
| Língua | Língua Gestual Portuguesa |
| Tipo | Vídeo |
| Número de Emissores[a] | 4 |
| Tamanho de Vocabulário | 34 |
| Número de Gestos Contínuos | 33 |
| Número de Gestos Isolados | 31 |
| Número de Gestos Contínuos | 2 |
| Número de Classes Nulas | 1 |
| Número Total de Vídeos[b] | 943 |
| Resolução do Vídeo | $1080 \times 1920$, 29,97 fps |
| Profundidade de Cor | 24 bpp, RGB |
| Formato do Vídeo | MOV |
| Espaço de Armazenamento[c] | 6,19 GB |
| Câmara | iPhone XR |

a) No campo **Número de Emissores**, existem quatro pessoas. No entanto, um das quatro pessoas não está presente em todas as classes, mas, nas classes em que aparece, a sua presença é idêntica às restantes.

b) No campo **Número Total de Vídeos** a distribuição de vídeos por classe não é uniforme, e algumas classes tiveram duas sessões de captura, duplicando a sua presença, como é possível observar na seguinte Tabela:

Tabela 4.3: Distribuição das classes na base de dados LGP-34

| **Classe** | **Nº de Vídeos** | **Classe** | **Nº de Vídeos** |
|---|---|---|---|
| Abril | 19 | Junho | 20 |
| Agosto | 29 | Laranja | 22 |
| Amarelo | 35 | Lilás | 35 |
| Azul | 40 | Maio | 17 |
| Bombeiro | 39 | Março | 13 |
| Branco | 37 | Medico | 39 |
| Castanho | 40 | Novembro | 21 |
| Cinzento | 18 | Outubro | 13 |
| Cor Laranja | 14 | Policia | 41 |
| Cor Rosa | 13 | Portugal | 44 |
| Dezembro | 16 | Preto | 40 |
| Dourado | 13 | Professor | 37 |
| Enfermeiro | 36 | Rosa | 40 |
| Fevereiro | 17 | Setembro | 18 |
| Janeiro | 11 | Verde | 20 |
| Juiz | 19 | Vermelho | 21 |
| Julho | 19 | P. Repouso | 87 |

c) No campo **Espaço de Armazenamento** todos os vídeos contêm áudio, sendo possível otimizar ainda mais este armazenamento.

### 4.2.3 SIGNUM

A base de dados SIGNUM é uma base de dados de DGS, e vem solucionar um problema existente em muitas base de dados de língua gestual, a dependência entre gesto-emissor, explicado na subsecção 4.3.2.

***(a)*** ***(b)*** ***(c)***

Figura 4.3: *Exemplo de um gesto isolado da base de dados SIGNUM. Tradução: "Maybe", Talvez*

Na base de dados SIGNUM, cada vídeo está identificado no seu nome, se é isolado ou contínuo. Sabe-se que, nos isolados só está presente um único gesto, e os mesmos são compostos por apenas 80 *frames*, e nos contínuos só é possível saber quantos gestos estão presentes, através da contagem do número de palavras de uma das etiquetas "*annot*", podendo encontrar um vídeo contínuo de dois a doze gestos, e de no máximo 390 *frames*.

Tabela 4.4: Informação das especificações da base de dados SIGNUM

| | |
|---|---|
| Nome | SIGNUM Database |
| Autor | Ulrich von Agris |
| Língua | Língua Gestual Alemã |
| Tipo | Vídeo |
| Número de Emissores[a] | 25 |
| Tamanho de Vocabulário | 450 |
| Número de Gestos Isolados | 450 |
| Número de Gestos Contínuos | 780 |
| Número Total de Vídeos | 33210 |
| Resolução do Vídeo | $776 \times 578$, 30 fps |
| Profundidade de Cor | 24 bpp, RGB |
| Formato da Imagem[b] | JPEG |
| Espaço de Armazenamento | 920 GB |
| Câmara | AVT Marlin F-046C |

a) No campo **Número de Emissores**, encontram-se estipulados 25 emissores, mas um dos emissores foi estabelecido como referência e por isso repetiu tudo três vezes, sendo na verdade 27 *performances*.

b) No campo **Formato da Imagem**, apesar do tipo de dados ser vídeo, o formato dos ficheiros armazenados são imagens. Isto acontece porque os vídeos foram segmentados

*frame* a *frame*, onde cada *frame* é armazenada como uma imagem em formato JPEG, numa pasta que representa o vídeo.

A base de dados SIGNUM contém anotações para cada vídeo que permitem saber qual o gesto que é realizado. Estas anotações oferecem sempre a tradução para duas línguas, inglês e alemão, e são diferente conforme o tipo de vídeo, isolado ou contínuo.

Nos vídeos de gestos isolados, vídeos compostos por um só gesto, as anotações estão disponíveis sob as etiquetas "*annot_eng*" e "*annot_deu*", onde ambas fornecem a tradução do gesto numa ou mais palavras, separadas pelo carácter "|". Nos vídeos de gestos contínuos, vídeos compostos por uma frase (ou seja, vários gestos) as etiquetas são "*annot_eng*" e "*annot_deu*", que traduzem cada gesto de forma independente e sequencial (*glosses*), "*transl_eng*" e "*transl_deu*", que traduzem a frase completa para inglês e alemão.

Tabela 4.5: Exemplo da anotação de um gesto isolado e de gestos contínuos

| Isolado | |
|---|---|
| annot_eng | TENT\|CAMPING |
| annot_deu | ZELT\|CAMPING |

| Contínuo | |
|---|---|
| annot_eng | CINEMA START WHEN? |
| annot_deu | KINO ANFANGEN WANN? |
| transl_eng | When does the film start? |
| transl_deu | Wann fängt das Kino an? |

A filmagem da base de dados foi realizada segundo algumas regras, que garantem a consistência entre as diferentes gravações. Todos os emissores foram filmados vestido com roupa escura e com fundo azul. A distância entre a câmara e o fundo foi estabelecida nos 2,50 metros e a distância entre o emissor e o fundo foi estabelecida nos 40 centímetros. A iluminação também é controlada e o ambiente de gravação está ilustrado na Figura 4.4.

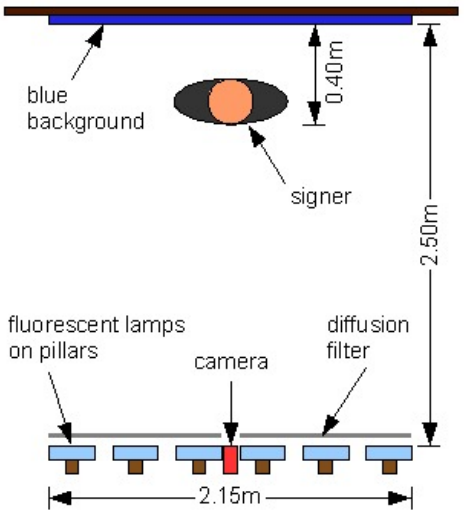


Figura 4.4: *Esquema da configuração de filmagem da base de dados SIGNUM. Imagem retirada de [44].*

Esta abordagem representada na Figura 4.4, embora garanta uma grande qualidade e consistência entre filmagens, acaba por não conseguir simular a utilização natural de uma aplicação onde não existe um ambiente controlado.

## 4.3 MediaPipe

A escolha da ferramenta MediaPipe foi motivada pela sua capacidade de reconhecimento visual, pela facilidade com que pode ser realizada a sua integração, e pela simplicidade do *output* que retorna.

O MediaPipe é uma ferramenta *open source* desenvolvida pela Google Research em 2019 [38], disponível em multi-plataformas, e permite ser executada em tempo real. Esta ferramenta disponibiliza dois tipos de abordagem, designados de **MediaPipe Framework** e **MediaPipe Solutions**, com o objetivo de facilitar o desenvolvimento de aplicações sensoriais (ligadas à visão ou audição). Estas duas abordagens permitem o desenvolvimento de algoritmos e modelos de Inteligência Artificial, utilizando os dados processados pela ferramenta.

A **MediaPipe Framework**, apresenta uma componente de baixo nível, concebida para a criação de *pipelines* personalizadas que processam os dados (vídeos ou áudios). A estrutura destas *pipelines*, chamada de *graphs*, consiste numa sequência modular de *nodes*, designados de *calculators*, que processam os dados, denominados de *packets*. Cada *calculator* realiza uma função específica, permitindo uma flexibilidade e personalização de cada *graph*. Esta *framework* permite, a quem a utilizar, criar a sua própria ferramenta de pré-processamento ou pós-processamento, sendo possível também focar-se num único tema, ou abranger diversos temas.

A **MediaPipe Solutions**, disponibiliza uma lista de modelos previamente criados, representando a abordagem de alto nível. Cada solução está disponível em múltiplas plataformas, e pode incluir um ou mais modelos, sendo que, quando múltiplos modelos estão disponíveis, estes variam em termos de desempenho e carga computacional. Entre as soluções disponíveis estão a deteção de objetos (*Object detection*), a classificação de texto (*Text classification*), a classificação de áudio (*Audio classification*), a deteção de landmarks na pose do corpo humano (*Pose landmark detection*), a detenção de *landmarks* na mão (*Hand landmark detection*), entre outras. É de notar também, que apesar de serem soluções previamente criadas, algumas permitem ser personalizadas.

No caso deste projeto, foi utilizada a solução *Hand landmarks detection*. Esta solução está disponível em múltiplas plataformas, e permite identificar e seguir (*tracking*), em vídeos ou imagens, de pontos das mãos, designados de *landmarks*. Estas *landmarks* são pontos representativos das coordenadas espaciais de elementos importantes da mão, como o pulso, as pontas dos dedos e as articulações dos dedos. Ao todo, a solução identifica 21 *landmarks* distintas, ilustradas na Figura 4.5.

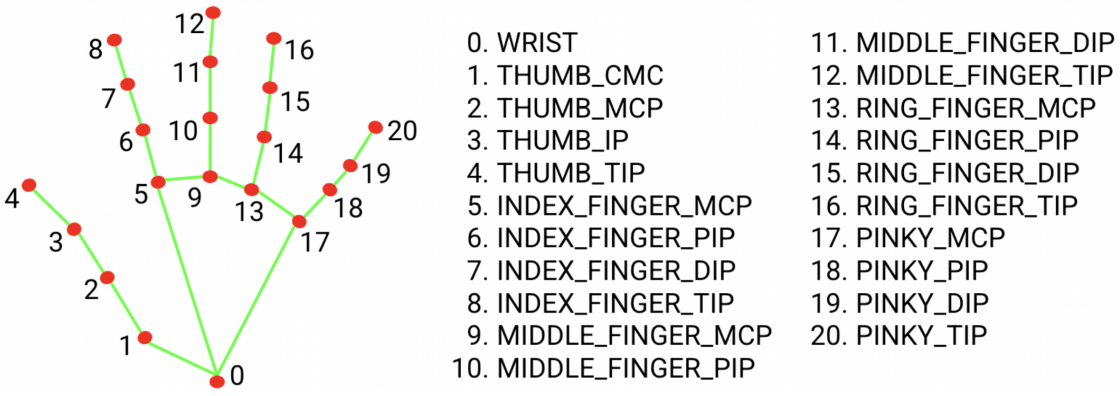


Figura 4.5: *Legenda das landmarks identificadas pela solução Hand landmarks detection. Imagem retirada de [45].*

A solução *Hand landmarks detection* contém duas componentes, que são utilizadas neste projeto: **Landmarks** e **Handedness**, e que desempenham tarefas distintas.

Na componente ***Landmarks***, as *landmarks* representam pontos no espaço que correspondem a coordenadas específicas dos elementos descritos na Figura 4.5. Estas coordenadas são normalizadas entre os valores 0 e 1 em relação às dimensões da *frame* a ser processada, e são apresentadas como coordenadas tridimensionais:

- X, refere-se à posição horizontal da *landmark* na *frame*, sendo que 0 corresponde ao limite esquerdo da imagem e 1 ao limite direito;
- Y, refere-se à posição vertical da *landmark* na *frame*, onde 0 corresponde ao limite superior da imagem e 1 ao limite inferior;
- Z, é uma estimativa da profundidade, onde o ponto de origem é dado pela coordenada Z da *landmark* 0 (pulso), e quanto menor for o valor, mais próximo a *landmark* encontra-se da câmara.

Embora o intervalo seja normalizado entre 0 e 1, é possível obter coordenadas superiores a 1 ou inferiores a 0. Esta situação deve-se ao modelo tentar prever a posição de todas as *landmarks*, mesmo quando os seus elementos encontram-se fora da *frame*, por exemplo, quando parte da mão encontra-se parcialmente fora do enquadramento.

Na componente ***Handedness*** é realizado o reconhecimento e distinção entre mão esquerda e mão direta, sendo esta distinção feita através de uma etiqueta. Esta funcionalidade será utilizada para organizar os dados adquiridos após o pré-processamento da base de dados.

### 4.3.1 Aplicação da ferramenta MediaPipe nas bases de dados

Antes de iniciar o treino dos modelos, é primeiro necessário realizar o pré-processamento dos dados através da solução *Hand landmarks detection* da ferramenta MediaPipe. A configuração desta solução apresenta uma diferença entre as base de dados, devido à capacidade

de *tracking*. Isto é, em vídeos a solução permite a utilização de *tracking*, o que facilita o reconhecimento contínuo das mãos, não sendo possível aplicar a mesma configuração em imagens. Tornando a Simples LGP Imagens, a única base de dados prejudicada.

Ao processar os dados, o MediaPipe retorna as *landmarks*, *frame* a *frame*, mesmo quando a opção de *tracking* se encontra ativa. Levando a ser necessário realizar o *nesting* dos *arrays* obtidos de cada *frame*, quando se trata de um vídeo.

Após obter-se os resultados da solução, organizou-se os dados da seguinte forma:

- Na concatenação dos dois *arrays* correspondentes às mãos esquerda e direita, estabeleceu-se que mão esquerda seria sempre a primeira, e a mão direta sempre a segunda. Ou seja, as primeiras 21 *landmarks* representam a mão esquerda, e as segundas 21 *landmarks* representam a mão direita;
- Por precaução, em casos onde não seja detetada uma ou ambas as mãos. A mão que não foi detetada é preenchida com zeros. Por exemplo, se a mão esquerda não for detetada, os primeiros 63 números (21 *landmarks*) serão zeros.

A Tabela seguinte apresenta os resultados finais do pré-processamento das diferentes bases de dados:

Tabela 4.6: Informação das bases de dados pré-processadas pela ferramenta MediaPipe

| Base de Dados | Armazenamento | | Tempo | Landmarks a zero[a] |
|---|---|---|---|---|
| | Landmarks | Original | | |
| LGP-9 | 2,58 MB | 4,33 GB | 5min 57s | 23,84% |
| LGP-34 | 81 MB | 6,19 GB | 1h 43min 54s | 6% |
| SIGNUM | 10,4 GB | 920 GB | 9d 17h 24 18s | 0.0005% |

a) O campo, **Landmarks a zero**, indica o número de *frames* onde não foi detetada qualquer mão, ou seja, todas as *landmarks* encontram-se a zero. Embora todas as bases de dados apresentem perdas. Apenas a LGP-9 apresenta perdas no número de elementos por classe. (Passando a sua distribuição a ser 174, 202, 232, 189, 138, 285, 246, 258, 135)

### 4.3.2 Gesto Independente do Emissor (Ângulos)

Citando o artigo [40], da base de dados SIGNUM:

> " *... Current systems for sign language recognition achieve excellent performance for signer-dependent operation. But their recognition rates decrease significantly if the signer's articulation deviates from the training data.*"

Existe um problema recorrente em algumas base de dados de língua gestual, designado de variabilidade interpessoal (*Interpersonal variability*), e que leva à queda do desempenho. Mesmo quando se trata do mesmo dialecto, o artigo [40], identifica que através da análise

do movimento da mão, é possível identificar que a variação entre diferentes emissores é significativamente maior do que a variação do próprio emissor, quando o mesmo repete o gesto.

No caso da base de dados SIGNUM, o conceito de Gesto Independente do Emissor, é aplicado através da existência de vinte e cinco emissores. Desta forma, permite-se que o algoritmo aprenda a reconhecer e traduzir os gestos, através de vinte e sete *performances* distintas.

Outro problema presente, desta vez causado pela ferramenta MediaPipe, e analisado por [26], é a sensibilidade que o algoritmo apresenta ao tamanho e à posição absoluta das mãos. Para além da variabilidade interpessoal, devido à variação da posição das mãos entre emissores. Este problema, também é afetado pela existência de um único cenário de captura na base de dados.

Como abordado anteriormente, na Figura 4.4, capturas onde o cenário é sempre o mesmo, levam a que o espaço de movimentos da mão, sejam praticamente dentro da mesma área. Carecendo de uma simulação realista, onde não existe um ambiente controlado, e onde a câmara não se encontra sempre à mesma distância do emissor.

Para solucionar este problema, o trabalho [26], apresenta o cálculo de ângulos entre os segmentos que unem as *landmarks*, como mais um passo no pré-processamento. Este passo, permite ignorar a informação espacial dada pela ferramenta MediaPipe, e trabalhar apenas com a verdadeira forma que a mão assume, ilustrada na Figura 4.6.

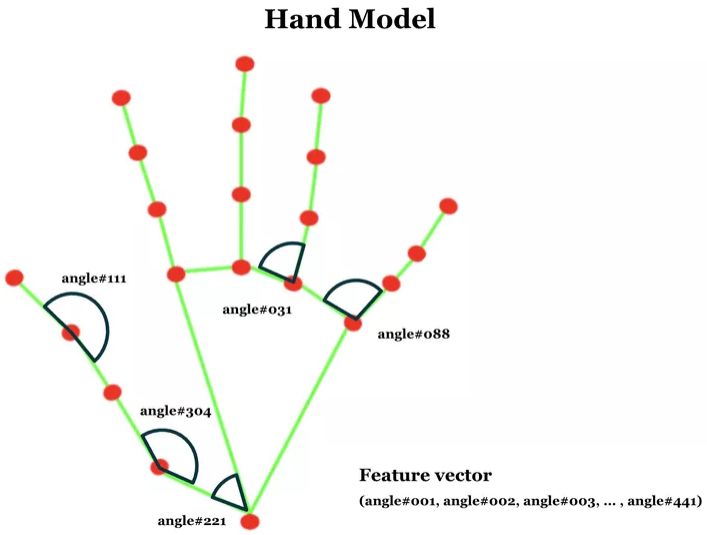


Figura 4.6: *Exemplos dos ângulos aplicados às landmarks do MediaPipe. Imagem retirada de [26].*

Como é possível observar na Figura 4.6, o cálculo de ângulos entre os segmentos que unem as *landmarks*, resulta num total de 441 ângulos, uma vez que são consideradas todas as combinações entre as 21 *landmarks* ($21 \times 21$). Ainda assim, este número pode ser reduzido,

pois a matriz de 441 números, é uma matriz simétrica, ou seja, pode-se trabalhar apenas com o triângulo superior obtendo como resultado final 231 ângulos ($\frac{21\times22}{2}$), ou 462 ângulos para duas mãos.

# 5 Classificação

> "*The road to wisdom?—Well, it's plain and simple to express: Err and err and err again but less and less and less.*"
>
> Piet Hein, *Grooks*

Neste capítulo é demonstrado os resultados obtidos na classificação de cada um dos métodos implementados, e é feita uma breve explicação de cada um. Na primeira secção, é implementado o algoritmo DTW. Na secções subsequentes é feita a implementação dos modelos de Inteligência Artificial seguintes: MLP, CNN e Transformers.

## 5.1 Dynamic Time Warping (DTW)

### 5.1.1 Classificação

Dynamic Time Warping ou DTW, é um algoritmo de alinhamento temporal utilizado na comparação entre duas sequências temporais [46], de duração diferente, ou que se encontram desfasadas. Este algoritmo lida com variações no tempo das sequências, ajustando dinamicamente a correspondência entre os instantes de tempo de cada sequência.

A utilização do algoritmo DTW neste projeto deve-se ao trabalho de Gabriel Guerin [26]. Um trabalho já mencionado anteriormente, que despertou um ponto de partida para a utilização da ferramenta MediaPipe, e que levou a testes iniciais com o algoritmo DTW. Contudo, este algoritmo não foi o foco deste projeto, tendo sido apenas uma implementação preliminar para testar a ferramenta MediaPipe.

O principal problema na utilização do algoritmo DTW, é o tempo necessário para obter os resultados. Isto, porque o algoritmo DTW necessita que cada sequência de teste seja comparada individualmente com todas as sequências no conjunto de treino, resultando num tempo de execução, que aumenta quanto maior for a base de dados. Este processo pode ser acelerado, com a utilização da biblioteca *fastdtw* [47], ainda assim os tempos de espera

mantiveram-se elevados.

Em termos de classificação, a biblioteca *fastdtw* oferece uma implementação simples e intuitiva. Nos seus parâmetros é dado uma sequência de teste e uma sequência de treino, obtendo um *path*, que serve para ilustrar o alinhamento realizado pelo algoritmo DTW, e uma distância, sendo esta distância a métrica utilizada para classificar o conjunto de teste.

Para preparar a classificação, a base de dados foi dividida em 80% treino e 20% teste, de seguida é percorrido sequência a sequência do conjunto de teste, comparando a sua distância a todas as sequências de treino. Por fim, para cada sequência de teste é armazenada a classe da sequência de treino mais próxima.

Tabela 5.1: Resultados das classificações do algoritmo DTW

| Base de Dados | Landmarks | | Ângulos | |
|---|---|---|---|---|
| | Taxa de Acerto | Tempo | Taxa de Acerto | Tempo |
| LGP-9 | 83% | 21 min 9 s | 91% | 3 h 25 min 49 s |
| LGP-34 | 80,31% | 17 min 33 s | 90% | 2 h 39 min 21 s |

A Tabela 5.1 apresenta os resultados das classificações. Como é possível observar, embora os resultados sejam bons, e note-se um melhoramento quando utilizado os ângulos. O tempo despendido na classificação de ambas as base de dados, em ambas as abordagens (landmarks e ângulos), começa a ser relativamente elevado em comparação com o tempo que se irá despender nos modelos de Inteligência Artificial. Não tendo sido sequer testada a base de dados SIGNUM, devido ao tempo que iria demorar.

### 5.1.2 Tipo de Reconhecimento e Aplicação

O algoritmo DTW, como implementado, compara duas sequências temporais já terminadas. Tornando impossível realizar uma tradução em tempo real, o tipo de reconhecimento escolhido, é o **Reconhecimento de Gestos Isolados**. Mais uma vez, este tipo de reconhecimento é o mais adequado para a classificação aplicada, embora talvez seja possível, através da análise do *path* chegar a outras conclusões, mas que fogem do âmbito deste projeto.

A implementação do algoritmo DTW, para a tradução de língua gestual para língua verbal, traria alguns problemas. Como é possível observar na Tabela 5.1, o tempo de classificação é elevado, devido à elevada carga computacional. Com 1487 imagens de treino e a aplicação de ângulos, pode-se concluir que uma única imagem de testes demora aproximadamente 33 segundos (cálculos disponíveis no Anexo I) a ser classificada. Este tempo, já por si elevado, só pode aumentar, quando a base de dados cresce, e a capacidade computacional do dispositivo diminui.

Uma potencial solução para baixar estes tempos de resposta seria a utilização de técnicas

como o "Representante mais próximo", que reduz o número de comparações necessárias. No entanto, a técnica utilizada neste projeto, "Um contra todos", garante os melhores resultados, e outras técnicas arriscam baixar a taxa de acerto.

## 5.2 Rede Neuronais de Perceptrão Multicamada

A Rede MLP apresenta a arquitetura mais simples entre os modelos de Inteligência Artificial implementados. Mas também apresenta facilmente o maior número de pesos, quando aplicada às bases de dados de vídeo. Isto porque cada coordenada de uma *landmark* corresponde a uma entrada na rede, e enquanto uma única *frame* apresenta apenas 126 pontos, um vídeo completo da base de dados SIGNUM pode apresentar até 49.140 pontos. Em comparação, este número equivale aproximadamente ao número de entradas necessárias para uma imagem de $\mathbf{128} \times \mathbf{128}$ pixels (cálculos disponíveis no Anexo II).

Embora as redes MLP consigam ser utilizadas para classificar imagens, existem arquiteturas mais adequadas para este tipo de tarefa. Ainda assim, foi explorada a capacidade das MLPs, tratando-se de redes simples de preparar e treinar.

Para as base de dados LGP-9 e LGP-34, tanto nas imagens como nos vídeos, foi implementado um modelo de duas camadas. A estrutura do modelo para esta base de dados, começa com uma camada de entrada, cujo o número de neurónios é determinado pelo produto entre o Número de *frames* (**T**) e Número de pontos das *landmarks* ou ângulos (**L**), e por sua vez, a camada de saída, é composta pelo Número de classes (**C**). A estrutura deste modelo pode ser observado na seguinte Figura 5.1.

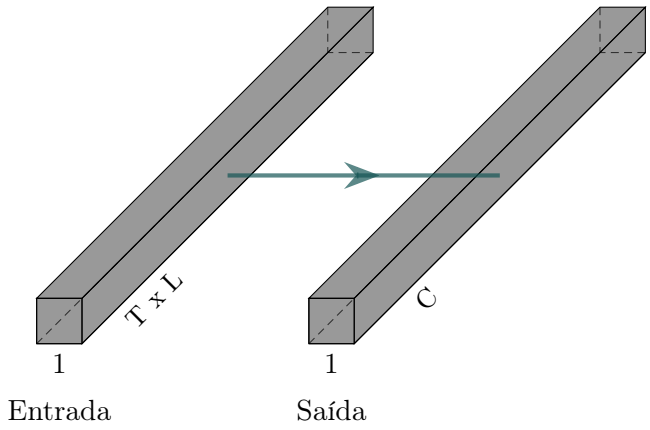


Figura 5.1: *Representação do modelo MLP implementado para a base de dados LGP. (**T**) Número de frames, (**L**) Número de pontos nas landmarks ou ângulos, (**C**) Número de classes. Desenhado através de PlotNeuralNet [48].*

Relativamente à base de dados SIGNUM, devido à quantidade de pesos que a rede iria apresentar, foi realizada outra abordagem. Em ambos os casos, *landmarks* e ângulos, foi aplicada uma camada escondida com 126 neurónios. Esta mudança no modelo devem-se à natureza das redes MLPs, pois a mesma configuração levaria a um número elevado de

pesos a ser treinado, com 60.296.007 (resultado de, $(126 \times 390) \times 1227 + 1227$) e 221.082.087 (resultado de, $(462 \times 390) \times 1227 + 1227$) pesos, respetivamente. Resultando no seguinte modelo, da Figura 5.2.

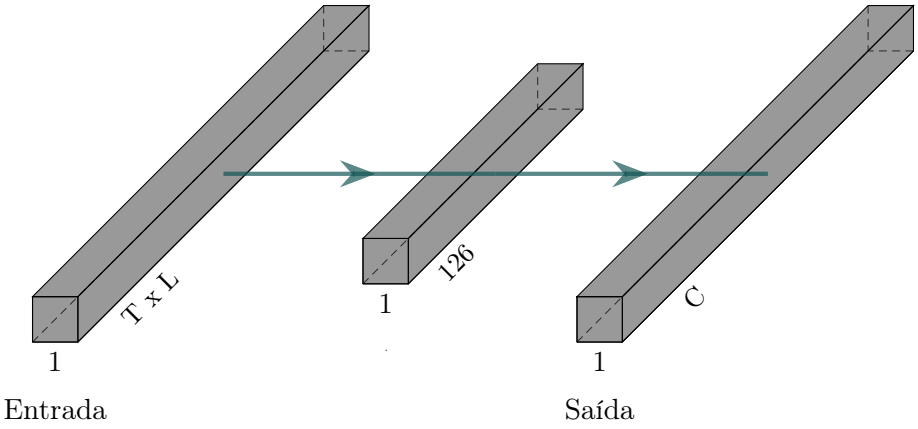


Figura 5.2: *Representação do modelo MLP implementado para a base de dados SIGNUM. (**T**) Número de frames, (**L**) Número de landmarks ou ângulos, (**C**) Número de classes. Desenhado através de PlotNeuralNet [48].*

O treino das bases de dados, LGP-9 e LGP-34, foi dividido em 60% treino, 20% validação e 20% teste, e o treino da base de dados SIGNUM, foi dividido em 80% treino, 10% validação e 10% teste. Em ambos os casos foi aplicada uma paragem prévia (*early stopping*) guiada pela perda (*loss*) do conjunto de validação. Ou seja, se durante o treino a *loss* do conjunto de validação não melhorar, durante dez épocas, o treino é parado antecipadamente. Após a realização do treino completo ou paragem prévia, é classificado o conjunto de teste, tendo-se obtido os seguintes resultados:

Tabela 5.2: Resultados do conjunto de teste no modelo MLP.

| Base de Dados | Landmarks | | | | Ângulos | | | |
|---|---|---|---|---|---|---|---|---|
| | Taxa de Acerto | Nº Pesos | Nº Época | Tempo de Treino | Taxa de Acerto | Nº Pesos | Nº Época | Tempo de Treino |
| LGP-9 | 93,55% | 1.143 | 614 | 17.3s | 97,04% | 7.947 | 287 | 9.79s |
| LGP-34 | 52,38% | 711.178 | 33 | 1.29s | 74,60% | 2.607.562 | 40 | 1min 2s |
| SIGNUM | 73,47% | 6.347.847 | 107 | 4min 39s | 83,44% | 22.858.887 | 28 | 2h 4min 51s |

**Nota**: O tempo da base de dados SIGNUM, quando a entrada é constituída por ângulos, não se trata apenas do treino, mas do cálculo dos ângulos e do treino.

Estes resultados indicam um desempenho significativamente superior para a classificação de imagens, particularmente quando utilizado ângulos. A mesma observação pode ser feita para todos os conjuntos, mas ao contrário das imagens, existe uma grande disparidade entre o número de pesos para vídeos.

### 5.2.1 Tipo de Reconhecimento e Aplicação

Semelhante à conclusão anterior, o tipo de tradução dos modelos MLPs, é **Reconhecimento de Gestos Isolados**. Isto porque a classificação pressupõem sempre o fim dos

dados de entrada.

Sobre a implementação do modelo MLP, para a tradução de língua gestual para língua verbal. É identificado, através da observação da Figura 5.2, a possibilidade de implementar a base de dados LGP-9, não para um fim comunicativo, mas para um fim educacional ou lúdico, onde o objetivo seria aprender e/ou identificar os números de 1 a 9 da Língua Gestual Portuguesa.

## 5.3 Redes Neuronais Convolucionais

As Redes CNNs, são redes compostas por camadas convolucionais [49], onde em cada camada são aplicados filtros (*kernels*), encarregues de detetar padrões em imagens, áudio, vídeos, entre outros. O treino destas redes envolve, treinar estes filtros, resultando num menor número de pesos, quando comparado às redes MLPs.

Ainda assim, as redes MLPs, continuam presente, pois a saída da última camada convolucional, é utilizada como entrada em camadas MLPs. Funcionando as CNNs, como um extrator de características para as redes MLPs.

Para a base de dados LGP-9, tratou-se as *landmarks* como imagens, com menos uma dimensão. Isto é, normalmente uma imagem costuma ser constituída pelas seguintes dimensões - Canal, Altura, Largura (*Channel*, *Height*, *Weight*). No caso das *landmarks* organizou-se as mesmas da seguinte forma - Axis, Landmarks - sendo Axis tratado como canal, e as Landmarks tratadas como altura ou largura de uma imagem. O modelo utilizado, é constituído por duas camadas de convolução unidimensional, separadas por um *Max Pooling*, e depois é realizado o *flat* dos dados para uma única camada MLP.

No caso das base de dados SIGNUM, existe uma instância de tempo, e por isso decidiu-se trabalhar com as *landmarks flat*, ficando com a seguinte dimensão - Landmarks, Tempo. O modelo utilizado para a base de dados SIGNUM é constituído por três camadas de convolução unidimensional, seguidas de um *flat* e quatro camadas MLP. Como é possível observar Figura 5.3:

Na base de dados LGP-34, adotou-se um modelo semelhante mas somente com duas camadas de convolução unidimensional e duas camadas MLP.

Relativamente ao treino, foram aplicadas as mesmas regras que nas redes MLPs. As bases de dados, LGP-9 e LGP-34, foram dividida em 60% treino, 20% validação e 20% teste, e a base de dados SIGNUM, foi dividida em 80% treino, 10% validação e 10% teste. Mais uma vez foi aplicada uma paragem prévia guiada pela *loss*. Após esta a paragem, ou realização do treino completo, é classificado o conjunto de teste, tendo-se obtido os seguintes resultados:

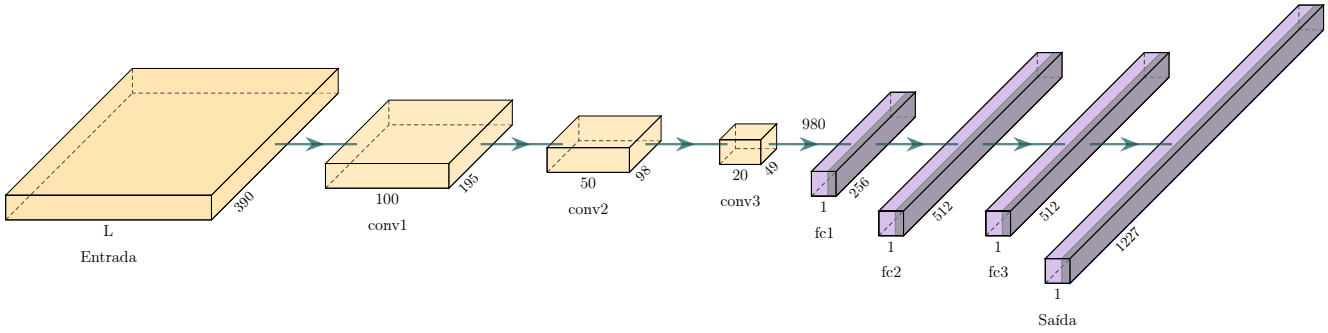


Figura 5.3: *Representação do modelo CNN implementado para a base de dados SIGNUM. (**L**) Número de pontos nas landmarks ou ângulos. Desenhado através de PlotNeuralNet[48].*

Tabela 5.3: Resultados do conjunto de teste no modelo CNN

| Base de Dados | Landmarks | | | | Ângulos | | | |
|---|---|---|---|---|---|---|---|---|
| | Taxa de Acerto | Nº Pesos | Nº Época | Tempo | Taxa de Acerto | Nº Pesos | Nº Época | Tempo |
| LGP-9 | 91,67% | 7.785 | 95 | 3.78s | 98,16% | 128.649 | 36 | 1.67s |
| LGP-34 | 71,43% | 34.894 | 185 | 3.55s | 85,71% | 85.294 | 59 | 1min 33s |
| SIGNUM | 81,57% | 1.332.158 | 73 | 3min 2s | 88,05% | 1.434.497 | 66 | 4h 22min 59s |

**Nota**: O tempo da base de dados SIGNUM, quando a entrada é constituída por ângulos, não se trata apenas do treino, mas do cálculo dos ângulos e do treino.

Comparando os resultados da Tabela 5.3, com os resultados da Tabela 5.2, concluí-se que o problema apresentado pela base de dados LGP-9, está praticamente resolvido. Ambas as abordagens apresentaram muito bons resultados, sendo talvez necessário um *fine tunning* final.

Ainda comparando ambas as tabelas. As base de dados com entrada de vídeo, apresentam melhorias tanto nos resultados, como na carga computacional, indicando as redes convolucionas como uma melhor abordagem.

Mais uma vez os ângulos continuam a apresentar melhores resultados, e desta vez os pesos encontram-se mais próximos (nos casos de vídeo), sendo possível concluir que os ângulos apresentam uma mais valia para este tipo de classificação.

### 5.3.1 Tipo de Reconhecimento e Aplicação

O método de classificação utilizado para ambas as arquiteturas, MLPs e CNNs, é o mesmo, e a única diferença encontra-se nas arquiteturas, tratando-se mais uma vez de **Reconhecimento de Gestos Isolados**.

Na implementação do modelo de CNNs, para tradução de língua gestual para língua verbal. Observa-se uma melhoria na taxa de acerto da LGP-34, e uma menor carga computacional, sendo possível conceptualizar uma aplicação para fins educativos ou lúdicos. Através da

aplicação de duas redes, uma que identifica números de 1 a 9 em Língua Gestual Portuguesa, e outra rede que identifica profissões, meses e cores.

## 5.4 Arquitetura Transformer

Introduzida no famoso artigo "*Attention Is All You Need*" [39] publicado em 2017, a arquitetura Transformer introduziu novas possibilidades no campo de Processamento de Linguagem Natural (NLP - *Natural Language Processingl*). Embora atualmente a sua popularidade se encontre sobretudo associada à geração de texto, focando-se mais numa abordagem *Decoder-only*. O seu propósito inicial era resolver um problema de tradução, mais especificamente, tradução de textos de inglês para francês e de inglês para alemão, onde ambos os lados da arquitetura eram utilizados *Encoder* e *Decoder*.

Atualmente, em NLP, as palavras são convertidas em *tokens*, que os Transformers processam em paralelo. Através de mecanismos de atenção que aprendem dependências contextuais ao longo da sequência, ajustando continuamente a informação que cada *token* representa. Este mecanismo permite a realização de tarefas de classificação, geração de texto e outras aplicações em NLP.

Além da NLP, existem outros domínios que também envolvem sequências de dados discretos, como a fala, música e gestos. Assim, é natural considerar a arquitetura Transformer para tarefas como a tradução de língua gestual, uma vez que esta envolve sequências temporais complexas, onde um ou mais gestos compõem uma sequência temporal de *tokens* que resumem informações como: movimento, posição, orientação e contexto específico.

Considerando a capacidade dos Transformers de lidar com contextos longos e sequências não lineares, não faz mais sentido continuar a utilizar as bases de dados LGP-9 e LGP-34, pois estas são de tamanho reduzido e constituídas apenas por gestos isolados. Deste modo, decidiu-se que, a partir deste momento, o projeto focará apenas na base de dados SIGNUM.

### 5.4.1 Arquitetura

O projeto desenvolvido tem por base a arquitetura de Andrej Karpathy [50], que disponibilizou um modelo do tipo *Decoder-only*, com o objetivo de educar e demonstrar o funcionamento do ChatGPT, e outras ferramentas idênticas. No entanto, este modelo é um *Decoder-only*, e especializa-se na previsão da palavra seguinte. Um dos trabalhos deste projeto é implementar o *Encoder*, conectá-lo ao que já se encontra desenvolvido e adaptá-lo ao contexto da tradução de língua gestual.

Para introduzir o *Encoder* na arquitetura, foi necessário implementar dois blocos de *Multi-Head Attention*. Um destes blocos foi inserido no *Encoder*, responsável por codificar os gestos recebidos, enquanto o outro bloco foi implementado no *Decoder*, e recebe dados

tanto do *Encoder* como do *Decoder*, servindo como ponte de ligação entre os dois modelos.

Após a implementação do *Encoder*, a arquitetura Transformer passou a apresentar dois modelos interligados por uma comunicação unidirecional, onde a saída do *Encoder* é fornecida como entrada para este novo bloco no *Decoder*. Consequentemente, a arquitetura no geral passou a apresentar duas entradas distintas. A entrada do *Encoder*, constituída por sequências temporais de *landmarks*. A entrada do *Decoder*, constituídas por *glosses* já convertidas em *tokens*. Ilustrado na Figura 5.4.

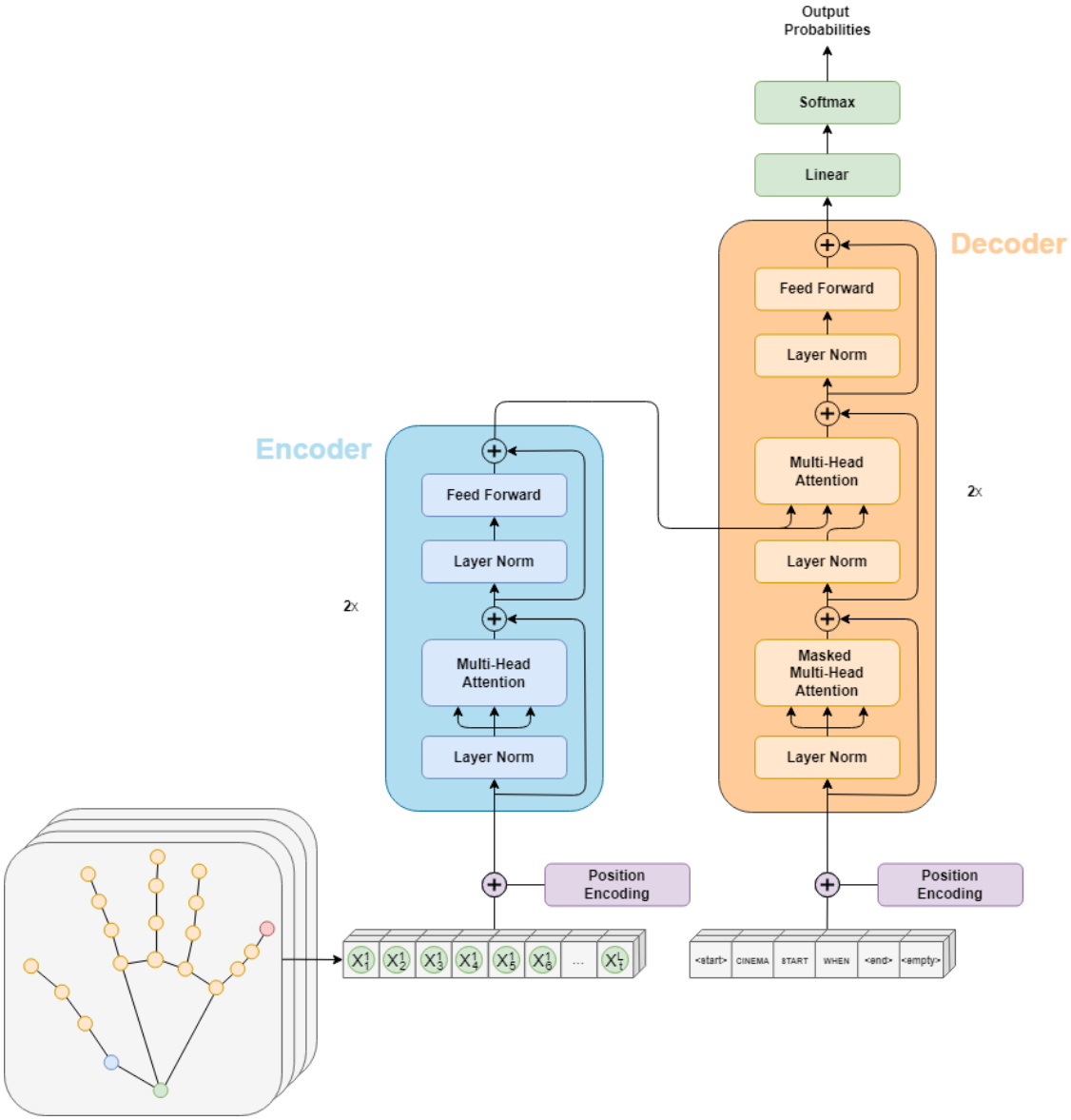


Figura 5.4: *Representação da arquitetura Transformer implementada. Referência de [39]*

Conforme representado na Figura 5.4, o resultado é uma variante do artigo original [39], designada de Pre-LN Transformer [51], pois apresenta as Camadas de Normalização (*Layer Normalization* ou *Layer Norm* na Figura 5.4) antes de cada Bloco de Atenção (*Multi-Head Attention* na Figura 5.4) e *Feed Forward*.

A adaptação do Transformer à tradução de língua gestual, levou a ser necessário modificar o método de entrada do *Encoder*. O *Encoder* recebe agora sequências temporais de *landmarks*, de dimensão (126, 390), onde 126 representa o total número de coordenadas de todas as *landmarks* e 390 representa o tempo em *frames*. Este formato de entrada permite que o *Encoder* processe diretamente as informações espaciais e temporais dos gestos, capturando nuances importantes para a tradução de língua gestual.

Para que o Transformer possa interpretar corretamente as sequências de entrada, é necessária a utilização de tabelas de embeddings (*embedding tables*) e codificação posicional (*position encoding).*

Encoder:

- *Embedding Table*: dimensão (126, 256), onde cada uma das 126 *landmarks* é representada num espaço vetorial de 256 dimensões.
- *Position Encoding*: dimensão (390, 256), permitindo que o modelo adicione informações sobre a posição de cada *frame* na sequência temporal.

Decoder:

- *Embedding Table*: dimensão $(453, 64)$, onde 453 corresponde ao número total de *tokens* (450 gestos isolados + 3 *tokens* especiais) e cada *token* é representado num espaço vetorial de 64 dimensões.
- *Position Encoding*: dimensão (14, 64), sendo 14 o tamanho máximo das sequências de *tokens* no *Decoder*.

A utilização de *embeddings* permite que o modelo capture relações semânticas entre as *landmarks* e os *tokens*, enquanto o *position encoding* assegura que a informação sobre a ordem das sequências seja preservada.

### 5.4.2 *Tokenizer*

Os exemplos apresentados a seguir serão baseados na anotação inglesa, contudo, o treino da arquitetura Transformer foi realizado com base nas anotações alemãs. Esta decisão deve-se à existência de palavras em alemão que, ao serem traduzidas para inglês, resultam numa única palavra, o que resultaria num conflito com a abordagem aplicada. As palavras são as seguintes:

- ”Wann” e ”Wenn” traduzem-se ambas para ”When”.
- ”Treffen” e ”Kennenlernen” traduzem-se ambas para ”Meet”.
- ”Ferien” e ”Urlaub” traduzem-se ambas para ”Holiday”.

A Tokenização é um passo fundamental no processamento de sequências em modelos de NLP, pois converte palavras num formato que o modelo consegue processar. No contexto deste projeto, o *Tokenizer* é somente aplicado nos dados de entrada do *Decoder*, ou seja, é aplicado nas frases alemãs, compostas por *glosses*. Estas frases não são a tradução final, mas sim a tradução direta de todos os gestos efetuados ao longo da sequência, e para serem processadas pelo *Decoder*, é necessário convertê-las em *tokens*.

Por exemplo, a frase "CINEMA START WHEN" (anotação em inglês: "*When does the film start?*" Tradução: "Quando começa o filme?") é composta por três gestos distintos. O *Tokenizer* divide esta sequência de gestos nos *tokens* "2, 3, 5". Assim, o descodificador do *Tokenizer*, ao receber a sequência "5, 3", retornará "WHEN START", sem qualquer divisão ou alteração na estrutura dos *glosses*.

Este sistema permite também que diferentes *glosses*, originados do mesmo gesto, ativem o mesmo número. Por exemplo, em Língua Gestual Alemã, o gesto "CINEMA" pode ser traduzido como "CINEMA" ou "FILM". Portanto, o codificador do *Tokenizer*, ao receber "FILM START WHEN", irá retornar precisamente os mesmos tokens "2, 3, 5". Na prática, o descodificador do *Tokenizer*, ao receber "2, 3, 5", retornará na verdade "[CINEMA, FILM] START WHEN".

No total, considerando que a base de dados SIGNUM possui 450 gestos isolados, o vocabulário do *Tokenizer* compreende 453 *tokens* (450 gestos isolados, mais 3 *tokens* especiais), e a correspondência de *tokens* para palavras é constituída por 586 chaves, devido a casos, como "CINEMA" que podem ser traduzidos em duas *strings*, "CINEMA" ou "FILM".

Além dos gestos isolados, foram acrescentados três *tokens* especiais para auxiliar na estruturação e compreensão das frases:

- **<start>**, indica o início de uma frase. Exemplo: "<start> CINEMA START WHEN";
- **<end>**, identifica o fim de uma frase. Exemplo: "<start> CINEMA START WHEN <end>";
- **<empty>**, designado de *padding token*, preenche as frases que não atingem o tamanho estabelecido pelo *block size*. Exemplo, se o *block size* for 6, a frase fica: "<start> CINEMA START WHEN <end> <empty>".

### 5.4.3 Treino

Após a implementação, do *Encoder* na arquitetura Transformer, e da aplicação do *Tokenizer* na frases de língua gestual. Foram realizados diversos treinos com o intuito de encontrar os melhores parâmetros. Aqui a abordagem adotada foi procurar o menor número de pesos possível, que ainda apresentasse bons resultados. Obtendo para ambos os modelos,

*Encoder* e *Decoder*, os parâmetros seguintes: 2 camadas (*layers*), e 8 cabeças de atenção (*heads*), e as dimensões de *embedding*, anteriormente mencionadas, de 256 e 64, respetivamente.

A utilização de *glosses* na tradução de todos os gestos, em vez da tradução em língua verbal, por exemplo, "CINEMA START WHEN", em vez de "When does the film start?", permite determinar se as palavras e frases produzidas estão corretas. Esta forma de avaliação não seria viável se se utilizasse a tradução em língua verbal, porque "When does the film start?" pode ser escrita de diversas formas, como "When is the film starting?", "When is the film going to start?" ou "When will the film start?", levando à necessidade da utilização de *Benchmarks*.

Para o treino da arquitetura Transformer decidiu-se dividir os conjuntos de treino, validação e teste de forma diferente. Em vez de se estabelecer percentagens fixas, optou-se por estabelecer que os primeiros 23 emissores (25 *performances*) fariam parte do conjunto de treino (92,59%), 1 emissor do conjunto de validação (3,70 %) e o último emissor do conjunto de teste (3,70 %). Esta divisão permite realizar um *early stopping* com base na classificação de uma pessoa que nunca foi vista durante o treino, e obter os resultados de classificação através de outra pessoa que nunca foi vista nem no treino, nem na validação.

Tabela 5.4: Resultados das classificações da arquitetura Transformer

| Base de Dados | Taxa de Acerto | | NºPesos | Nº Iterações | Tempo |
|---|---|---|---|---|---|
| | Frase | Palavra | | | |
| SIGNUM | 92.93% | 98.53% | 1.951.621 | 21.000 | 2h 52min 35s |

Interpretando os resultados da Tabela 5.4, verifica-se que, no conjunto de teste, 92.93% das frases estão a ser corretamente traduzidas e 98.53% das palavras em todas as frases estão a ser corretamente avaliadas. Relembrando que a base de dados SIGNUM é composta por dois tipos de dados, gestos contínuos e gestos isolados, decidiu-se dividir estes resultados.

Gestos Contínuos:

- Total de frases: 780
- Total de frases corretas: 762 (**97.69%**)
- Total de palavras corretas: 6496 de 6543 (**99.28%**)

Gestos Isolados:

- Total de gestos: 450
- Total de gestos corretos: 381 (**84.67%**)
- Total de palavras corretas: 1281 de 1350 (**94.89%**)

**Nota**: É avaliada a classificação dos *tokens* especiais <start> e <end>. Embora ocorra em ambos os casos, torna-se mais evidente nos gestos isolados, onde existem somente 450 gestos, mas compara-se 1350 *tokens*.

Ao comparar os resultados entre tipos de gestos, não só neste treino, como também nos vários treinos anteriormente realizados para ajustar os pesos do modelo, observa-se que os gestos isolados apresentam sempre resultados significativamente inferiores aos gestos contínuos. Este problema poderá surgir devido a variados fatores.

Os gestos isolados são naturalmente mais curtos, que os gestos contínuos, seja na entrada de *landmarks*, seja na entrada de *tokens*. Sendo maioritariamente compostos por zeros na entrada do *Encoder*, e por *padding tokens* (<empty>) na entrada do *Decoder*. Sendo possível a perda de informação em ambos os lados, quando é necessário comprimir os dados, durante o processamento do modelo.

Outro problema, poderá surgir devido à capacidade que a arquitetura Transformer apresenta. Podendo o modelo não estar a traduzir, mas sim a memorizar, perante apenas os 1230 exemplos disponíveis. Existindo outros fatores para além da variabilidade interpessoal, como por exemplo o próprio contexto.

Embora este dois problemas sejam apenas suposições, e não tenha sido explorado uma forma de as comprovar, foi testado e realizado um método diferente, que vem combater ambos os fatores. Pois no caso da base de dados SIGNUM, não se encontra identificado o início e fim de cada gesto, trabalhando-se apenas com dois tipos de dados: frases completas (gestos contínuos), ou palavras (gestos isolados). Não existindo uma forma de treinar o modelo com frases incompletas e simular uma tradução em tempo real.

Este desafio surgiu da dúvida "Será possível simular uma tradução em tempo real, através da aplicação de sequências temporalmente segmentadas?", e apresenta também possíveis soluções aos dois problemas anteriores (Compressão e Memorização).

### 5.4.4 Extensão do Vocabulário

Confrontado com a dúvida de tradução em tempo real e segmentação temporal, decidiu-se utilizar os gestos isolados para construir novas frases. Embora esta abordagem apresente a desvantagem da captura dos gestos isolados, começarem e terminarem sempre com o emissor em posição de repouso, não permitindo simular uma conversa real em língua gestual. Existe a vantagem de possibilitar a criação de mais frases além das que já existem.

Na construção de novas frases foi necessário identificar diversos verbos, nomes comuns, locais, comidas, animais, entre outros, criando um total de 65.590 novas frases. Não é possível determinar quantas destas frases encontram-se gramaticalmente corretas, portanto,

garantiu-se sempre a utilização de referências de frases já existentes. Por exemplo, na frase "I SHOES NEW NEED" (Anotação em inglês: "*I need new shoes*", Tradução: "Eu preciso de novos sapatos"), substituiu-se o nome comum "SHOES" por "UMBRELLA", "BOOK", "TELEVISION" entre outros, e também se alterou o pronome "I" por "YOU", "TEACHER", "WAITER", entre outros, respeitando a construção frásica original.

Com as novas frases, foi possível construir sequências utilizando emissores diferentes, como frases em que cada gesto é realizado por um emissor distinto. Contribuindo para a abordagem do problema da variabilidade interpessoal.

Esta segmentação temporal também permite simular uma tradução em tempo real, ao estabelecer-se um cenário em que o dispositivo captura a 30 *frames* por segundo, armazenando-as em memória, e a cada 30 *frames* o modelo classifica todas as *frames* armazenadas até surgir um *token* especial. Esta abordagem assegura que o modelo aguarda pelo fim da execução completa do gesto antes de proceder à classificação. Por exemplo, mesmo tendo 60 das 80 *frames* do gesto "WHEN", a frase correta ainda é processada como "<start> CINEMA START".

Para abranger casos onde a frase se encontra incompleta, foi adicionado um quarto *token* especial, <inc> . Este *token* representa uma frase incompleta, como por exemplo, "<start> CINEMA START <inc>".

Por fim, o Transformer anteriormente treinado teve também que ser alterado. Decidiu-se reduzir o número de cabeças (*heads*) de 8 para 2, e a utilização exclusiva de frases segmentadas resultaou num aumento do *block size* para 960, levando ao aumento do número de pesos. Onde os resultados obtidos foram os seguintes:

Tabela 5.5: Resultados das classificações da arquitetura Transformer, para simulação de tempo real

| Base de Dados | Taxa de Acerto | | NºPesos | Nº Iterações | Tempo |
|---|---|---|---|---|---|
| | Frase | Palavra | | | |
| SIGNUM | 81,77% | 96,52% | 2.097.541 | 23500 | 2h 27min 33s |

Os resultados apresentados na Tabela 5.5, são mais baixos que os resultados da Tabela 5.4, no entanto, e semelhante ao modelo anterior, decidiu-se dividir as frases por tamanhos. Desta vez, não existindo gestos isolados ou contínuos, a divisão foi realizada através do número de palavras (*tokens*) em cada frase.

Tabela 5.6: Amostra dos resultados de 10 *batches* criados aleatoriamente, para diferentes números de palavras na simulação de tempo real

| Nº Palavras | Nº Sequências | Taxa de Acerto | |
|---|---|---|---|
| | | Frase | Palavra |
| 0 | 72 | 100.00% | 100.00% |
| 1 | 101 | 99.01% | 99.67% |
| 2 | 62 | 91.94% | 96.77% |
| 3 | 120 | 83.33% | 94.83% |
| 4 | 133 | 81.95% | 95.36% |
| 5 | 144 | 86.11% | 97.22% |
| 6 | 225 | 82.67% | 96.06% |
| 7 | 176 | 78.98% | 97.22% |
| 8 | 150 | 80.67% | 97.73% |
| 9 | 46 | 80.43% | 97.43% |
| 10+ | 45 | 64.44% | 95.83% |

**Nota:** As frases com 0 palavras surgem quando o primeiro gesto ainda não se encontra completo, e o modelo apenas tem como output o *token* especial <start>.

Na Tabela 5.6, o número de palavras é constituído tanto por frases que contêm, por exemplo, 6 palavras, como também por frases, que contêm mais palavras, e que aleatoriamente foram cortadas para 6 palavras.

Os resultados das Tabelas 5.5 e 5.6 permitem concluir que é possível realizar uma tradução em tempo real, e onde anteriormente havia sido posto em causa dois possíveis problemas, na tradução entre gestos isolados e gestos contínuos, a distribuição dos resultados presente na Tabela 5.6 demonstram que através de dados mais abrangentes é possível obter melhores resultados.

Isto porque, já não se pode supor um problema de compressão, devido a distribuição da taxa de acerto, nem se pode supor uma memorização do contexto, porque desta vez não existem apenas 2 ou 3 exemplos para um tipo de frase, como foi demonstrado no exemplo da frase "*I need new shoes*".

### 5.4.5 Tipo de Reconhecimento e Aplicação

Ao trabalhar com a arquitetura Transformer e uma base de dados que contém mais do que um gesto por vídeo, é possível realizar um **Reconhecimento Contínuo de Língua Gestual**. A questão que se coloca, é se é possível executar a tradução em tempo real ou se é necessário trabalhar com vídeos pré-gravados.

No caso do treino que originou os resultado da Tabela 5.4, apenas se trabalha com frases completas ou gestos isolados, sendo impossível simular uma tradução em tempo real. Já na extensão de vocabulário, representada pelos resultados da Tabela 5.5, é possível realizar uma tradução em tempo real, pois a montagem personalizada de frases, permite simular

frases incompletas e lidar com gestos que ainda não terminaram.

Diferente dos outros modelos, decidiu-se implementar uma aplicação que utilizasse os modelos Transformer anteriormente treinados. O processo de implementação e ferramentas utilizadas encontra-se descrito no próximo Capítulo 6.

# 6 Implementação

*"... Not all those who wander are lost ..."*

J.R.R. Tolkien, *The Fellowship of the Ring*

Este capítulo descreve o processo de implementação de uma aplicação, através dos modelos Transformer treinados, destacando as ferramentas utilizadas e os vários componentes presentes na *interface* da aplicação.

## 6.1 Objetivo e Ferramentas

Através da comparação dos resultados obtidos nas diversas classificações realizadas, torna-se evidente que a arquitetura Transformer, não só apresenta melhores resultados, como também possibilita o reconhecimento contínuo, algo que as restantes arquiteturas não permitem. Deste modo, decidiu-se desenvolver uma aplicação multiplataforma destinada a implementar e testar os dois modelos Transformer previamente treinados.

A aplicação tem como propósito, exercer a função de canal num sistema de comunicação. Isto é, pretende-se que a aplicação seja capaz de capturar a mensagem emitida em língua gestual pelo emissor, traduzir essa mensagem e transmiti-la ao recetor através de áudio ou texto. Apresentando uma comunicação unidirecional, pois é pretendido traduzir a mensagem de língua gestual para língua verbal.

No entanto, apesar da capacidade dos modelos Transformer treinados, estes não são capazes, por si só, de capturar vídeos nem processar dados de entrada que não sejam *landmarks*. Além disso, os modelos não realizam a tradução para língua verbal, mas sim para *glosses*, nem possibilitam a transmissão da mensagem através de áudio. Sendo necessário utilizar um conjunto de ferramentas e tecnologias externas aos modelos para o desenvolvimento completo da aplicação.

Diversas são as ferramentas que possibilitam a criação de uma aplicação multiplataforma, que permita a captura de vídeos, a tradução para língua verbal e a transmissão da mensagem ao recetor através de áudio ou texto.

Entre as várias possibilidades, existe apenas uma ferramenta obrigatória, a ferramenta **MediaPipe**, devido à sua utilização no pré-processamento dos dados e treino. Esta ferramenta, semelhante à sua função anterior, irá ser responsável por extrair as *landmarks* das mãos, dos vídeos capturados pela aplicação.

O Open Neural Network Exchange ou **ONNX**, é uma ferramenta *open source* para a representação de modelos de aprendizagem profunda, que permite a passagem e leitura entre diferentes *frameworks*, ferramentas e compiladores [52]. Através do ONNX, consegue-se importar os pesos dos modelos Transformer treinados e exporta-los para a aplicação.

A biblioteca **Kivy**, e o seu *widget* complementar KivyMD, são um software *open source* desenvolvido em Python, que possibilita o desenvolvimento da *interface* gráfica da aplicação [53]. Esta biblioteca serve para criar aplicações multiplataforma, e foi utilizada para desenhar toda a *interface* gráfica do utilizador (GUI - *Graphical User Interface*) mais à frente demonstrada.

Devido à saída, dos modelos Transformer, ser em formato de *glosses*, foi necessário um segundo processamento que realizasse a correspondência entre os *glosses* obtidos e as frases desejadas em língua verbal. Através do ***fine-tuning* do modelo GPT-4o mini**, modelo este disponibilizado a 18 Julho de 2024 e a possibilidade do seu *fine-tuning* disponibilizado dias mais tarde [54], é possível o modelo aprender e traduzir de *glosses* para língua verbal.

Embora seja possível estabelecer uma comunicação somente através de texto, decidiu-se, adicionalmente, oferecer um *feedback* auditivo com a integração da biblioteca Python **edge_tts**. Esta biblioteca utiliza o serviço Microsoft Edge text-to-speech [55], e após ser definida uma voz, a biblioteca edge_tts recebe uma frase e retorna o aúdio correspondente, sendo apenas necessário o reproduzir.

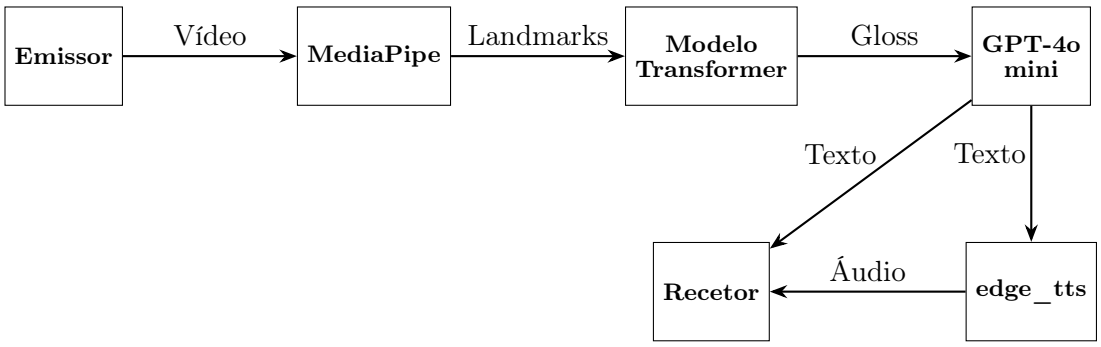


Figura 6.1: *Representação do fluxo dos dados e função de cada ferramenta.*

## 6.2 Design

A aplicação, como mencionado anteriormente, foi desenvolvida através da utilização da biblioteca Kivy e KivyMD, e foi estruturada em três componentes principais, ilustradas na Figura 6.2.

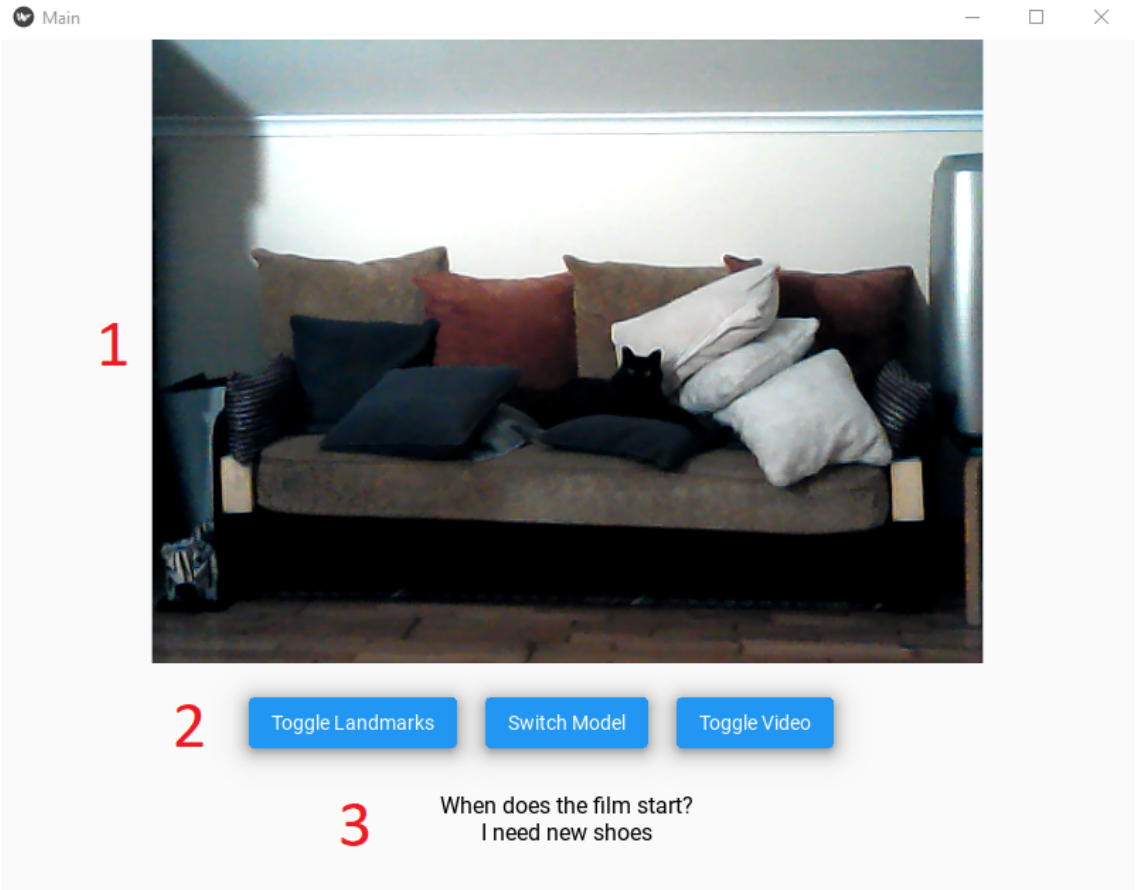


Figura 6.2: *Design da aplicação desenvolvida*

**1)** Feedback da Câmara, onde é fornecido ao utilizador uma visualização em tempo real do que está a ser capturado pela câmara, do dispositivo a ser utilizado. Isto permite ao utilizador ajustar a posição ou enquadrar o emissor, quando necessário.

Esta componente também permite visualizar vídeos pré-gravados, como por exemplo, vídeos da base de dados SIGNUM, e a alteração entre captura em tempo real ou pré-gravado é feito através do botão da componente a seguir explicada.

**2)** Botões, cada botão apresenta uma função diferente e são de uma natureza mais técnicas que as restantes componentes.

Toggle Landmarks, este botão serve para ativar ou desativar a visualização das *landmarks* na primeira componente. Este botão é sobretudo uma verificação se a ferramenta MediaPipe está a ser capaz de extrair as *landmarks* da imagem que está a ser visualizada.

Switch Model, dado que se pretende testar dois modelos Transformer, este botão permite ao utilizador alternar entre eles, modificando o modelo que irá classificar as *landmarks* extraídas ao longo de um determinado período de tempo pelo MediaPipe.

Toggle Video, este botão permite alterar o que está a ser visualizado na primeira componente, entre captura em tempo real ou vídeos pré-gravados.

**3)** Legenda da Tradução, este componente é uma *label* que apresenta a tradução ao longo da captura. Contém até três linhas de texto, que são preenchidas de cima para baixo, descartando sempre a mais antiga, e a quebra entre linhas é realizada através do *token* especial <end>, que está presente no *output* do modelo, mas que nunca é exibido.

Devido aos modelos serem diferentes, e um deles ser capaz de traduzir língua gestual a cada 30 *frames*, a linha mais recente pode nem sempre ser composta por uma frase em língua verbal, mas por *glosses*. Isto porque, o modelo que traduz a cada 30 *frames* apresenta e atualiza a tradução em tempo real, sendo a passagem feita para língua verbal, somente depois de se obter o *token* especial <end>.

Todo este desenvolvimento foi considerado um passo extra, onde a prioridade encontra-se na utilização e funcionamento dos modelos, e não no *design*. A biblioteca Kivy permitiu manter este desenvolvimento na linguagem de programação Python, e com um *design* simples e modular, sem ser necessário explorar outras *frameworks* ou ferramentas como React, Flutter, entre outras.

## 6.3 Tradução de Gloss para Língua Verbal

A tradução de *glosses* para língua verbal, é por si só um desafio e poderia ser o tema principal de um projeto. Assim sendo, será apenas explicado possíveis soluções para este problema e a razão pela qual se escolheu o *fine-tuning* do modelo GPT-4o mini.

O ***fine-tuning*** envolve ajustar um modelo pré-treinado para uma tarefa específica, neste caso, tradução de *glosses* para a língua verbal. Atualmente, existem vários modelos capazes de concretizar esta tarefa, como Llama, Mistral, GPT, entre outros.

No caso de modelos de pesos partilhados publicamente como Llama e Mistral, o *fine-tuning* é realizado numa máquina pessoal, condicionando a mesma a um poder computacional elevado. Por isso, optou-se pelo GPT-4o mini, devido à capacidade de realizar o *fine-tuning* remotamente, à facilidade do processo feito através da OpenAI Plataform e à data, e específico para o modelo GPT-4o mini, ser permitido o seu fine-tuning gratuito. Ainda assim, a OpenAI Plataform requer a preparação e organização dos dados da seguinte forma:

Listagem 6.1: Estrutura requisita pela OpenAI Plataform, exemplo de duas linhas em jsonl.

```
{
    "messages": [
        {"role":"system", "content": Prompt de sistema},
        {"role":"user", "content": Glosses},
        {"role":"assistant", "content": Frase em língua verbal}
    ]
}
{
    "messages": [
        {"role":"system", "content": "You are a Sign Language Translator"},
        {"role":"user", "content": "[['CINEMA', 'FILM'], 'START', 'WHEN']"},
        {"role":"assistant", "content": "When does the film start?"}
    ]
}
...
```

**Nota:** Cada entrada deve estar escrita numa só linha. Nesta demonstração, para facilitar a leitura dos exemplos, decidiu-se quebrar esta regra.

Esta organização foi feita automaticamente através de código Python, onde os *glosses* foram preenchidos da forma como são preparados para treino, e as frases de língua verbal foram retiradas na sua integra, das anotações disponíveis da base da dados SIGNUM.

Uma das desvantagens desta utilização é não existir qualquer controlo sobre palavras que o modelo nunca tenha visto durante o seu *fine-tuning*, podendo apenas combater este problema através de mais exemplos, ou de dados representativos de todos os casos possíveis. Outras desvantagens desta utilização é necessidade de conectividade à Internet, devido aos serviços da OpenAI não se encontrarem disponíveis localmente, e os custos após o *fine-tuning* do modelo, pois a utilização do modelo *fine-tuned* implicada pagar por cada *token*.

Outra solução seria ***prompt engineering***, que consiste em construir e elaborar instruções específicas para guiar o modelo a produzir as respostas desejadas. Esta abordagem, é semelhante ao processo anterior de *fine-tuning*, pois requer um modelo pré-treinado, ligação à Internet e pode envolver custos, dependendo do modelo utilizado.

Semelhante ao *fine-tuning*, é necessário preparar previamente os dados. No entanto, *prompt engineering* implica explorar várias *prompts*, por exemplo:

> Atua como um tradutor profissional de língua gestual para língua verbal.
> A tua tarefa é converter frases escritas na sintaxe da língua gestual (*glosses*) em frases gramaticalmente corretas em inglês, assegurando que o significado original seja mantido. A língua gestual frequentemente omite palavras funcionais

como artigos, preposições e auxiliares, além de utilizar uma ordem de palavras diferente do inglês padrão. Ao traduzir, deves:

- Reorganizar a ordem das palavras para se adequar ao inglês convencional.
- Adicionar palavras conforme necessário.
- Garantir que os tempos verbais se encontrem corretos.
- Preservar o sentido e tom da frase original.

Exemplos:

"CINEMA START WHEN?" é traduzido para "When does the film start?"

"I SHOES NEED NEW" é traduzido para "I need new shoes."

...

Utilizando estes exemplos como referência, traduz a seguinte frase:

Uma das vantagens e desvantagens desta abordagem, encontra-se na quantidade de *prompts* que se pode construir. Em *prompt engineering* não existe a *prompt* correta, pois várias podem ser capazes de realizar a mesma tarefa. No entanto *prompts* semelhantes podem obter resultados diferentes. Tomando como exemplo, a *prompt* anteriormente partilhada e a frase "Preciso de um novo automóvel", a passagem de "tradutor profissional" para "tradutor" (no inicio da *prompt*) pode resultar numa maior probabilidade do surgimento da palavra "carro" em vez de "automóvel". Sendo necessário testar várias *prompts*, e por vezes alterar pequenos detalhes nessas mesmas *prompts*.

Por fim, a última opção tomada em conta, envolve treinar **outro modelo Transformer**, dedicado a receber *glosses* e a produzir a frase desejada em língua verbal, ou então, trabalhar com as *landmarks* e produzir diretamente a frase em língua verbal.

Embora nenhuma das abordagens tenha sido testada, ambas foram ponderadas, com a utilização de *transfer learning* no lado do *Decoder*. Pois, *transfer learning* poderia mitigar problemas que surgissem com palavras que o modelo nunca teria visto inseridas em certas frases ou contexto, e traria um conhecimento gramatical que nunca iria ser possível obter através do treino de raiz do *Decoder* com somente 1230 frases.

Embora seja possível supor várias desvantagens, esta é uma das opções que só após a sua realização seria possível identificar concretamente as suas vantagens e desvantagens. No início do projeto, optou-se por um modelo Transformer que recebe *landmarks* e produz *glosses*, em vez de língua verbal. Sendo esta implementação um passo extra, optou-se por escolher a opção mais simples, e aproveitar o período gratuito do *fine-tuning* do modelo GPT4o-mini, em vez de treinar mais um Transformer.

## 6.4 Resultado

Após o desenvolvimento da aplicação, não foi possível realizar testes com um falante natural de DGS. Ainda assim, através da aprendizagem de algumas palavras e utilização do primeiro modelo, é possível obter a tradução correta das frases testadas.

Por outro lado, ao utilizar-se os vídeos pré-gravados do conjunto de teste, vídeos estes realizados por profissionais, os resultados obtidos mantêm-se iguais aos da Tabela 5.4.

Na utilização do segundo modelo, os testes sem vídeos pré-gravados, não são viáveis. Como referido anteriormente, a montagem de frases realizada para este modelo não vai de encontro a uma conversa natural em língua gestual, e devido à precisão necessária de executar cada gesto em 80 *frames*, torna-se impossível testar o mesmo. Sendo apenas possível realizar testes a este segundo modelo através da montagem de vídeos pré-gravados. Obtendo-se resultados semelhantes aos anteriores.

Por fim, o desenvolvimento da aplicação provou a eficácia dos modelos Transformer na tradução de língua gestual. Onde o desenvolvimento de uma aplicação, com integração de outras ferramentas externas, apenas vem complementar os modelos em si. Como por exemplo, o *fine-tuning* do modelo GPT-4o mini, que permitiu passar *glosses* para frases em língua verbal, enriquecendo a comunicação pretendida.

# 7 Conclusões

> "O fim duma viagem é apenas o começo doutra."
>
> José Saramago, Viagem a Portugal

Neste capítulo são apresentadas as principais conclusões do trabalho e trabalhos futuros a explorar.

## 7.1 Conclusão

Neste trabalho comprovou-se a utilidade da ferramenta MediaPipe ao longo dos vários treinos, não só através dos resultados obtidos, como também através da sua facilidade de implementação e possibilidade de criar redes neuronais com menos pesos que o normal.

Além disso, foi utilizada a base de dados SIGNUM para treinar diversos modelos de Inteligência Artificial, definindo-se o tipo de reconhecimento de cada modelo. Concluiu-se que as redes MLP e CNN apresentam capacidades de ser implementadas em aplicações básicas de tradução isolada, e em ambos os casos sempre com melhores resultados, quando utilizado a representação por ângulos. Além disso, uma arquitetura Transformer foi adaptada para a tradução da língua gestual, permitindo alcançar o objetivo deste trabalho: o Reconhecimento de Língua Gestual Contínua.

Após o treino e classificação da arquitetura Transformer, decidiu-se estender o vocabulário da base dados SIGNUM, e simular um ambiente de comunicação em tempo real, podendo mais uma vez voltar a treinar e classificar o mesmo. Mais tarde, decidiu-se utilizar estes dois modelos para desenvolver uma aplicação capaz de traduzir língua gestual para língua verbal, obtendo sucesso nessa implementação.

Todas as abordagens, à exceção do algoritmo DTW e redes MLPs para vídeos, demonstram ótimos resultados e qualquer uma é viável para uma implementação, dada as necessidades e objetivos da aplicação desejada.

## 7.2 Trabalho Futuro

Este trabalho iniciou-se com o objetivo de traduzir Língua Gestual Portuguesa para Língua Portuguesa. Contudo, existiram dificuldades na obtenção de uma base de dados de Língua Gestual Portuguesa, portanto um dos objetivos futuros consiste na criação de uma base de dados de Língua Gestual Portuguesa.

Para além de uma base de dados de LGP, existem outros requisitos que podem ser realizados, para criar uma base de dados mais complexa que a base de dados SIGNUM, tanto a nível de vocabulário, como a nível de *labelling*, e tudo através de uma captura, ainda que profissional, mais representativa de um ambiente quotidiano.

Num outro plano mais técnico, após a exploração do passado das línguas gestuais, surge a possibilidade de investigação em torno dos queremas, tal como apresentados por William Stokoe. Esta abordagem implica a identificação dos parâmetros que compõem um gesto, os queremas. Em vez de se identificar o gesto por inteiro, o objetivo seria decompor em queremas todos os gestos. Resultando em:

| | |
|---|---|
| Número de mãos | 1 |
| Configuração | 1 |
| Orientação | Palma para dentro |
| Movimento | Estático |
| Localização | Nível do peito |
| Gesto (Tradução) | 1 |

| | |
|---|---|
| Número de mãos | 1 |
| Configuração | 1 |
| Orientação | Palma para dentro |
| Movimento | Dinâmico cima - baixo |
| Localização | Queixo |
| Gesto (Tradução) | Branco |

| | | |
|---|---|---|
| Número de mãos | 2 | |
| Configuração | Dominante | n |
| Orientação | Dominante | Palma para dentro |
| Movimento | Dominante | Dinâmico alternado |
| Localização | Dominante | Nível do peito |
| Configuração | Não Dominante | 1 |
| Orientação | Não Dominante | Palma para cima |
| Movimento | Não Dominante | Estático |
| Localização | Não Dominante | Nível do peito |
| Gesto (Tradução) | Andar | |

Esta abordagem, embora implique um trabalho prévio mais exigente, devido à necessidade de identificação dos queremas que compõe um gesto. Permite a implementação e tradução de qualquer língua gestual, sendo primeiro necessário "apenas" desenvolver uma *Lookup Table* ou uma Máquina de Estados (dependendo da abordagem e das necessidades) de toda uma língua. Idealizando até, a possível divisão de um modelo, que é constituído por dois componentes, um destinado à identificação de queremas, e outro responsável pelo reconhecimento de gestos através dos queremas identificados.

Outro ponto a considerar é a utilização da ferramenta MediaPipe para extrair junto das *landmarks* das mãos, as *landmarks* da face e corpo. Embora este projeto se tenha concentrado exclusivamente nas *landmarks* das mãos, é importante relembrar que existem línguas gestuais, onde as expressões faciais e corporais, desempenham um papel importante na sua língua. Por exemplo, a anteriormente mencionada, Língua Gestual Indo-Paquistanesa, onde uma expressão facial pode alterar o significado do gesto.

A investigação nesta área ainda apresenta vários caminhos e possibilidades, e todas elas levam a experiências novas.

# Bibliografia


[1] Dicionário Priberam da Língua Portuguesa. *língua natural*. Acedido a 31 de Julho de 2024. 2024. URL: https://dicionario.priberam.org/lÃŋngua%20natural (ver p. 5).

[2] Étienne Bonnot de Condillac. *An Essay on the Origin of Human Knowledge: Being a Supplement to Mr. Locke's Essay on the Human Understanding*. Scholars' Facsimiles e Reprints, 1756. URL: https://archive.org/details/anessayonorigin00condgoog (ver p. 5).

[3] H. Barnard. *Tribute to Gallaudet: A Discourse in Commemoration of the Life, Character and Services of the Rev. Thomas H. Gallaudet, Ll.D.* With an appendix containing history of deaf-mute instruction and institutions, and other documents. Hartford: Brockett, Hutchinson & Co., 1852. URL: https://openlibrary.org/books/OL7053247M/Tribute_to_Gallaudet (ver p. 6).

[4] C. de L'Épée. *Institution des sourds et muets, par la voie des signes méthodiques: Ouvrage qui contient le Projet d'une Langue Universelle, par l'entremise des Signes naturels assujettis à une Méthode, Volume 1*. Chez Nyon l'ainé, 1776. URL: https://books.google.pt/books?id=BeIUAAAAQAAJ (ver pp. 6, 8).

[5] H. Lane. *When the Mind Hears: A History of the Deaf*. Publicado originalmente em 1984. Vintage, 2010. URL: https://play.google.com/store/books/details?id=gZ9e2B-WfU0C (ver pp. 6, 7).

[6] J. Massieu, R. Sicard, L. Clerc, J. Sievrac, A. de Ladébat e A. de Ladébat. *Recueil des définitions et réponses les plus remarquables de Massieu et Clerc, sourds-muets, aux diverses questions quileur ontété faites dans les séances publiques de M. l'Abbé Sicard, à Londres: auquel on a joint l'alphabet manuel des sourds-muets, le discours d'ouverture de M. l'Abbé Sicard, et une lettre explicative de sa méthode*. Imprimé par Cox et Baylis, 1815. URL: https://books.google.pt/books?id=hv8SAAAAIAAJ (ver p. 6).

[7] J. S. Long. *The Sign Language, a Manual of Signs*. Washington, D.C.: Press of Gibson bros, 1910. ISBN: 9781331920946. URL: https://archive.org/details/signlanguagemanu00long (ver p. 6).

[8] A. d. Costa. *No Minho*. Lisboa: Imprensa Nacional, 1874. URL: https://archive.org/details/nominho00costuoft (ver pp. 6, 8, 9).

[9] W. C. Stokoe Jr. "Sign language structure: An outline of the visual communication systems of the American deaf". Em: *Journal of deaf studies and deaf education* 10.1 (2005), pp. 3–37. URL: https://academic.oup.com/jdsde/article/10/1/3/361306 (ver p. 7).

[10] E. Klima e U. Bellugi. *The Signs of Language*. Harvard University Press, 1979. ISBN: 9780674807969. URL: https://books.google.pt/books?id=WeBOn6N8PJ8C (ver p. 7).

[11] S. K. Liddell e R. E. Johnson. "American sign language: The phonological base". Em: *Sign language studies* 64.1 (1989), pp. 195–277. URL: https://muse.jhu.edu/pub/18/article/507116/summary (ver p. 7).

[12] R. Battison. "Phonological deletion in american sign language". Em: *Sign language studies* 5.1 (1974), pp. 1–19. URL: https://muse.jhu.edu/pub/18/article/507140/summary (ver pp. 7, 8).

[13] U. Zeshan. *Sign Language in Indo-Pakistan: A Description of a Signed Language*. John Benjamins Publishing Company, 2000. ISBN: 9789027225634. URL: https://books.google.pt/books?id=TY-dmc1h5OQC (ver p. 8).

[14] P. A. Borg. *Golpe de Vista sobre a Necessidade, Valor e Importancia de hum Estabelecimento de Educacao para os Surdos-Mudos e Cegos*. Biblioteca Nacional de Portugal. Lisboa: Impressão da Viúva Neves e Filhos, 1828. URL: https://purl.pt/38915 (ver p. 8).

[15] J. Prawitz. *Pär Aron Borg. Svenskt biografiskt lexikon*. Acedido a 13 de Agosto de 2024. URL: https://sok.riksarkivet.se/sbl/artikel/17981 (ver p. 8).

[16] J. C. da Cunha. *História do Instituto dos Surdos-Mudos e Cegos de Lisboa, desde a sua Fundação até á sua incorporação na Casa Pia*. Rua do Arco do Bandeira n. 117. Lisboa: Typograp. de Filippe Nery, 1835. URL: https://purl.pt/38567 (ver pp. 8, 9).

[17] M. do Céu Garcia dos Reis Loureiro Alves. *Educação especial e modernização escolar: Estudo histórico-pedagógico da educação de surdos-mudos e de cegos*. 2012. URL: http://hdl.handle.net/10451/8687 (ver p. 9).

[18] A. dos Santos. *O Ensino dos Surdos-Mudos em Portugal: comunicação feita à Sociedade de estudos pedagógicos*. Comunicação feita à Sociedade de Estudos Pedagógicos. Lisboa, 1913. URL: https://www.google.pt/books/edition/O_ensino_dos_surdos_mudos_em_Portugal/jvd8HAAACAAJ?hl=pt-PT (ver p. 9).

[19] Diário do Governo, nº133. de 17 de Junho de 1870. URL: https://digigov.cepese.pt/pt/pesquisa/listbyyearmonthday?ano=1870&mes=6&tipo=a-diario&filename=1870/06/17/D_0133_1870-06-17&pag=1&txt= (ver p. 9).

[20] Diário do Governo, nº264. de 20 de Novembro de 1894. URL: https://digigov.cepese.pt/pt/pesquisa/listbyyearmonthday?ano=1894&mes=11&tipo=a-diario&filename=1894/11/20/D_0264_1894-11-20&pag=4&txt= (ver p. 9).

[21] República Portuguesa. *Constituição da República Portuguesa, de acordo com a Lei n.º 1/97 de 20 de Setembro, Artigo 74.º, alínea h.* https://dre.pt/dr/legislacao-consolidada/decreto-aprovacao-constituicao/1976-34520775-49472775. 1997 (ver p. 9).

[22] A. Krizhevsky, I. Sutskever e G. E. Hinton. "ImageNet Classification with Deep Convolutional Neural Networks". Em: *Advances in Neural Information Processing Systems*. Ed. por F. Pereira, C. Burges, L. Bottou e K. Weinberger. Vol. 25. Curran Associates, Inc., 2012. URL: https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf (ver p. 11).

[23] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson et al. "Sam 2: Segment anything in images and videos". Em: *arXiv preprint arXiv:2408.00714* (2024). URL: https://ai.meta.com/research/publications/sam-2-segment-anything-in-images-and-videos/ (ver p. 11).

[24] S. Tamura e S. Kawasaki. "Recognition of sign language motion images". Em: *Pattern Recognition* 21.4 (1988), pp. 343–353. ISSN: 0031-3203. DOI: https://doi.org/10.1016/0031-3203(88)90048-9. URL: https://www.sciencedirect.com/science/article/pii/0031320388900489 (ver p. 12).

[25] N. Tanibata, N. Shimada e Y. Shirai. "Extraction of hand features for recognition of sign language words". Em: *International conference on vision interface*. 2002, pp. 391–398. URL: https://www.researchgate.net/profile/Yoshiaki-Shirai-3/publication/2904731_Extraction_of_Hand_Features_for_Recognition_of_Sign_Language/links/53feb7990cf21edafd151e69/Extraction-of-Hand-Features-for-Recognition-of-Sign-Language.pdf (ver pp. 12–15).

[26] G. Guerin. *Sign Language Recognition - using MediaPipe & DTW*. https://data-ai.theodo.com/blog-technique/sign-language-recognition-using-mediapipe. Acedido a 22 de Julho de 2024. 2022 (ver pp. 12–15, 27, 29).

[27] G. Li, H. Tang, Y. Sun, J. Kong, G. Jiang, D. Jiang, B. Tao, S. Xu e H. Liu. "Hand gesture recognition based on convolution neural network". Em: *Cluster Computing* 22 (2019), pp. 2719–2729. URL: https://link.springer.com/article/10.1007/s10586-017-1435-x (ver pp. 12–15).

[28] I. Papastratis, K. Dimitropoulos, D. Konstantinidis e P. Daras. "Continuous sign language recognition through cross-modal alignment of video and text embeddings in a joint-latent space". Em: *IEEE Access* 8 (2020), pp. 91170–91180. URL: https://ieeexplore.ieee.org/abstract/document/9090828 (ver pp. 12–15).

[29] L. Pigou, S. Dieleman, P.-J. Kindermans e B. Schrauwen. "Sign language recognition using convolutional neural networks". Em: *Computer Vision-ECCV 2014 Workshops: Zurich, Switzerland, September 6-7 and 12, 2014, Proceedings, Part I 13*. Springer. 2015, pp. 572–578. URL: https://link.springer.com/chapter/10.1007/978-3-319-16178-5_40 (ver pp. 12–15).

[30] J. Huang, W. Zhou, H. Li e W. Li. "Sign language recognition using 3d convolutional neural networks". Em: *2015 IEEE international conference on multimedia and expo (ICME)*. IEEE. 2015, pp. 1–6. URL: https://ieeexplore.ieee.org/abstract/document/7177428 (ver pp. 12–15).

[31] N. C. Camgoz, O. Koller, S. Hadfield e R. Bowden. "Sign Language Transformers: Joint End-to-End Sign Language Recognition and Translation". Em: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020. URL: https://arxiv.org/abs/2003.13830 (ver pp. 12–15).

[32] M. Boháček e M. Hrúz. "Sign Pose-Based Transformer for Word-Level Sign Language Recognition". Em: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops*. 2022, pp. 182–191. URL: https://openaccess.thecvf.com/content/WACV2022W/HADCV/html/Bohacek_Sign_Pose-Based_Transformer_for_Word-Level_Sign_Language_Recognition_WACVW_2022_paper.html (ver pp. 12–15).

[33] A. Halder e A. Tayade. "Real-time vernacular sign language recognition using mediapipe and machine learning". Em: *Journal homepage: www. ijrpr. com ISSN* 2582 (2021), p. 7421. URL: https://www.researchgate.net/profile/Akshit-Tayade-2/publication/369945035_Real-time_Vernacular_Sign_Language_Recognition_using_MediaPipe_and_Machine_Learning/links/643605da20f25554da283357/Real-time-Vernacular-Sign-Language-Recognition-using-MediaPipe-and-Machine-Learning.pdf (ver pp. 12, 14, 15, 17).

[34] C. Luna-Jiménez, M. Gil-Martín, R. Kleinlein, R. San-Segundo e F. Fernández-Martínez. "Interpreting sign language recognition using transformers and MediaPipe landmarks". Em: *Proceedings of the 25th International Conference on Multimodal Interaction*. 2023, pp. 373–377. URL: https://dl.acm.org/doi/abs/10.1145/3577190.3614143 (ver pp. 12, 14, 15).

[35] T. Pryor e N. Azodi. *Lemelson-MIT Program*. https://lemelson.mit.edu/award-winners/thomas-pryor-and-navid-azodi. Acedido a 26 de Agosto de 2024. 2016 (ver pp. 12, 16).

[36] Indriani, M. Harris e A. S. Agoes. "Applying Hand Gesture Recognition for User Guide Application Using MediaPipe". Em: *Proceedings of the 2nd International Seminar of Science and Applied Technology (ISSAT 2021)*. Atlantis Press, 2021, pp. 101–108. ISBN: 978-94-6239-451-3. DOI: 10.2991/aer.k.211106.017. URL: https://doi.org/10.2991/aer.k.211106.017 (ver pp. 12–15, 17).

[37] Q. Feng, C. Yang, X. Wu e Z. Li. "A smart TV interaction system based on hand gesture recognition by using RGB-D Sensor". Em: *Proceedings 2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC)*. IEEE. 2013, pp. 1319–1322. URL: https://ieeexplore.ieee.org/document/6885271 (ver pp. 12–15).

[38] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee et al. "Mediapipe: A framework for building perception pipelines". Em: *arXiv preprint arXiv:1906.08172* (2019). URL: https://arxiv.org/abs/1906.08172 (ver pp. 15, 17, 24).

[39] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser e I. Polosukhin. *Attention Is All You Need*. 2023. arXiv: 1706.03762 [cs.CL]. URL: https://arxiv.org/abs/1706.03762 (ver pp. 15, 35, 36).

[40] U. von Agris e K.-F. Kraiss. "SIGNUM Database: Video Corpus for Signer-Independent Continuous Sign Language Recognition". Inglês. Em: *7th International Conference on Language Resources and Evaluation (LREC 2010). Proceedings of the LREC2010 4th Workshop on the Representation and Processing of Sign Languages: Corpora and Sign Language Technologies*. Ed. por P. Dreuw, E. Efthimiou, T. Hanke, T. Johnston, G. Martínez Ruiz e A. Schembri. Valletta, Malta: European Language Resources Association (ELRA), mai. de 2010, pp. 243–246. URL: https://www.sign-lang.uni-hamburg.de/lrec/pub/10006.pdf (ver pp. 15, 17, 18, 26).

[41] U. Von Agris, M. Knorr e K.-F. Kraiss. "The significance of facial features for automatic sign language recognition". Em: *2008 8th IEEE international conference on automatic face & gesture recognition*. IEEE. 2008, pp. 1–6 (ver p. 15).

[42] U. Von Agris, C. Blomer e K.-F. Kraiss. "Rapid signer adaptation for continuous sign language recognition using a combined approach of eigenvoices, MLLR, and MAP". Em: *2008 19th International Conference on Pattern Recognition*. IEEE. 2008, pp. 1–4 (ver p. 15).

[43] H. Talk. *Hand Talk - Comunicando de Forma Inclusiva*. https://www.handtalk.me/br/home/. Acedido a 26 de Agosto de 2024. 2024 (ver p. 16).

[44] U. von Agris. *SIGNUM Database for Signer-Independent Continuous Sign Language Recognition*. https://www.phonetik.uni-muenchen.de/forschung/Bas/SIGNUM/. Acedido a 10 de Setembro de 2024. 2013 (ver p. 23).

[45] G. AI. *MediaPipe Hand Landmarker*. https://ai.google.dev/edge/mediapipe/solutions/vision/hand_landmarker. Acedido a 4 de Setembro de 2024. 2024 (ver p. 25).

[46] H. Sakoe e S. Chiba. "Dynamic programming algorithm optimization for spoken word recognition". Em: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 26 (1978), pp. 159–165. URL: https://api.semanticscholar.org/CorpusID:17900407 (ver p. 29).

[47] S. Salvador e P. Chan. "FastDTW: Toward accurate dynamic time warping in linear time and space". Em: *Intelligent Data Analysis* 11.5 (2007), pp. 561–580. URL: https://cs.fit.edu/~pkc/papers/tdm04.pdf (ver p. 29).

[48] H. Iqbal. *PlotNeuralNet: Latex code for drawing neural networks*. https://github.com/HarisIqbal88/PlotNeuralNet. 2018 (ver pp. 31, 32, 34, 67, 68).

[49] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard e L. D. Jackel. "Backpropagation applied to handwritten zip code recognition". Em: *Neural computation* 1.4 (1989), pp. 541–551. URL: https://ieeexplore.ieee.org/abstract/document/6795724 (ver p. 33).

[50] A. Karpathy. *Neural Networks: Zero to Hero - Video Lectures*. Acedido a 2 de Setembro de 2024. 2022. URL: https://github.com/karpathy/ng-video-lecture (ver p. 35).

[51] R. Xiong, Y. Yang, D. He, K. Zheng, S. Zheng, C. Xing, H. Zhang, Y. Lan, L. Wang e T. Liu. "On layer normalization in the transformer architecture". Em: *International Conference on Machine Learning*. PMLR. 2020, pp. 10524–10533 (ver p. 36).

[52] *ONNX: Open Neural Network Exchange*. Acedido a 20 de outubro de 2024. 2017. URL: https://onnx.ai (ver p. 46).

[53] K. Organization. *Kivy*. Acedido a 20 de outubro de 2024. 2011. URL: https://kivy.org (ver p. 46).

[54] OpenAI. *GPT-4O Mini: Advancing Cost-Efficient Intelligence*. Acessado a 21 de outubro de 2024. 2024. URL: https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/ (ver p. 46).

[55] Rany2. *edge-tts*. Acessado a 21 de outubro de 2024. 2024. URL: https://github.com/rany2/edge-tts (ver p. 46).

# I Cálculos DTW

Dados:

- Número total de sequências de teste: 372 (20% de 1.859, total de imagens na bd)
- Tempo total de classificação: 3 horas, 25 minutos e 49 segundos

Conversão para segundos:

$$3 \text{ horas} = 3 \times 3.600 = 10.800 \text{ segundos}$$

$$25 \text{ minutos} = 25 \times 60 = 1.500 \text{ segundos}$$

$$49 \text{ segundos} = 49 \text{ segundos}$$

Soma:

$$10.800 + 1.500 + 49 = 12.349 \text{ segundos}$$

Divisão do tempo total pelo número de sequências de teste:

$$\frac{12.349}{372} \approx 33,19 \text{ segundos por conjunto de teste}$$

**Resultado**: O tempo para classificar um exemplo é aproximadamente **33,19 segundos**.

# II Cálculos da entrada de Rede MLP

Dados:

- Número de pontos numa *landmark*: 126
- Número máximo de *frames* na bd SIGNUM: 390

Número total de entradas numa Rede Neuronal:

$$126 \text{ pontos} \times 390 \text{ frames} = 49.140 \text{ pontos}$$

Tratar os 49.140 pontos como se fosse uma imagem RGB:

$$\frac{49.140 \text{ pontos}}{3 \text{ canais}} = 16.380 \text{ pixels}$$

Achar uma resolução:

$$\sqrt{16.380 \text{ pixels}} = 127.98 \approx 128$$

**Resultado**: O maior vídeo presente na base de dados SIGNUM, após processado pela ferramenta MediaPipe, apresenta aproximadamente a mesma informação que uma imagem RGB de $\mathbf{128} \times \mathbf{128}$.

# III Representação para outros casos do modelo CNN

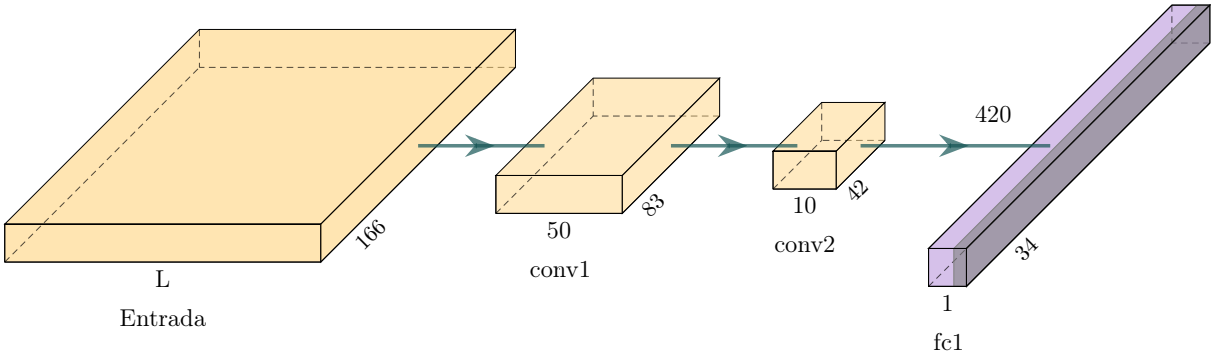


Figura III.1: *Representação do modelo CNN implementado para a base de dados Simples LGP Vídeos. (**L**) Número de pontos nas landmarks ou ângulos. Desenhado através de PlotNeuralNet [48].*

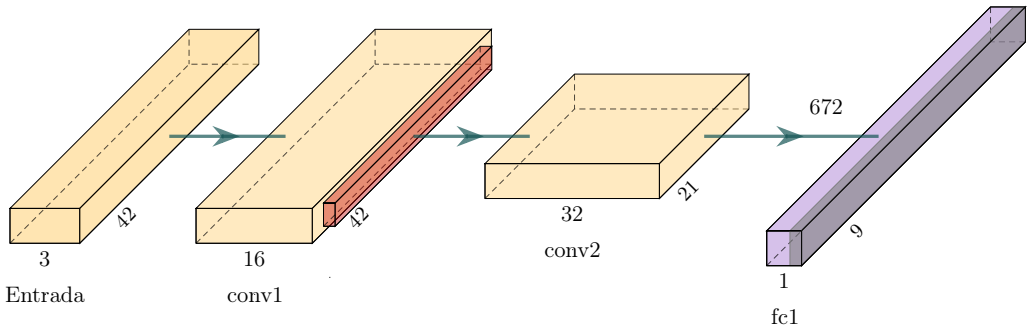


Figura III.2: *Representação do modelo CNN implementado para a base de dados Simples LGP Imagens Landmarks. Desenhado através de PlotNeuralNet [48].*

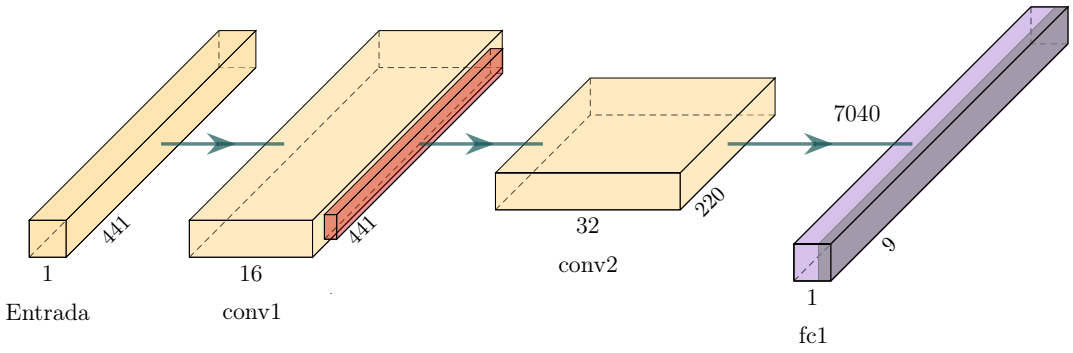


Figura III.3: *Representação do modelo CNN implementado para a base de dados Simples LGP Imagens Ângulos. Desenhado através de PlotNeuralNet [48].*