



INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA (ISEL)

DEPARTAMENTO DE ENGENHARIA ELETRÓNICA E DE
TELECOMUNICAÇÕES E COMPUTADORES (DEETC)

LEIM

LICENCIATURA EM ENGENHARIA INFORMÁTICA E MULTIMÉDIA
UNIDADE CURRICULAR DE PROJETO

Anotação de Eventos Sonoros em Vídeo

Carina Fernandes (45118)

Orientadores

Professor [Doutor] Joel Paulo

Professor [Doutor] Paulo Trigo

Professor Paulo Vieira

Setembro, 2022

Resumo

O desempenho de um atleta, numa determinada modalidade desportiva, melhora quando este é acompanhado de uma perspetiva externa no decurso da sua atividade desportiva. Neste sentido, o atleta pode ser monitorizado por um treinador para atingir melhores resultados. Como complemento, é possível registar em vídeo e analisar posteriormente o desempenho do atleta. Neste sentido, é vantajoso o desenvolvimento de uma ferramenta capaz de realizar essa análise externa.

A ferramenta desenvolvida neste projeto permite extrair e reconhecer batidas de bola em raquete (i.e. períodos em que ocorra uma maior troca de bolas) com base no vídeo da atividade desportiva do atleta (em treinos ou competições de padel /ténis). O processamento é realizado sobre o áudio extraído de um vídeo, e tem como objetivo a extração de padrões identificativos das batidas de bola, com o auxílio de técnicas de aprendizagem automática, baseadas em séries temporais.

No final, através de uma aplicação *web*, são visualizados os resultados repetantes aos instantes em que ocorrem batidas de bola, dando uma perspetiva mais abrangente e objetiva do desempenho do atleta. A ferramenta identifica corretamente cerca de 85% dos eventos em análise, sendo necessários alguns ajustes para melhorar o processo de reconhecimento.

Os ensaios são realizados no Laboratório de Áudio e Acústica do ISEL, LAA.

Palavras-chave: algoritmos de software de deteção do som, algoritmos de software de inteligência artificial.

Abstract

In a given sport, the athlete's performance improves when he is supervised from an external perspective. In this scenario, the athlete can be supervised by a coach in order to achieve better results, or as a complement, it is possible to video record and analyze his performance afterwards. Thus, it is advantageous to develop a tool capable of performing this external analysis.

The tool developed in this project, allows extracting and recognizing relevant events (i.e., periods when a greater exchange of balls occurs) based on the video of the athlete's sport activity (in the padel/tennis sport activity). The processing is performed on the audio extracted from a video, and aims at extracting identifying patterns of the ball hits, with the help of automatic learning techniques based on time series.

At the end, through a *web* application, the results relative to the instants in which ball hits occur are visualized, giving a more comprehensive and objective perspective of the athlete's performance. The tool correctly identifies about 85% of the events under analysis, with some adjustments needed to improve the recognition process.

The tests are carried out at ISEL's Audio and Acoustics Laboratory, LAA.

Keywords: sound detection software algorithms, artificial intelligence software algorithms.

Agradecimentos

Escrever aqui eventuais agradecimentos ...

Eventual texto de dedicatória . . .

. . . mais texto,

. . . e o fim do texto.

Índice

| | |
|--|----------|
| Resumo | i |
| Abstract | iii |
| Agradecimentos | v |
| Índice | ix |
| Lista de Tabelas | xi |
| Lista de Figuras | xiii |
| 1 Introdução | 1 |
| 1.1 Motivação | 1 |
| 1.2 Principais Contributos | 2 |
| 1.3 Validação e Testes | 2 |
| 1.4 Estrutura do Relatório | 2 |
| 2 Fundamentos | 5 |
| 2.1 Conceitos de Processamento de Sinal | 5 |
| 2.1.1 Amostragem | 5 |
| 2.1.2 Transformada de Fourier | 6 |
| 2.1.3 Características extraídas do áudio | 6 |
| 2.2 Conceitos de Aprendizagem Automática | 7 |
| 2.2.1 <i>Dataset</i> | 8 |
| 2.2.2 Desiquilíbrio no <i>Dataset</i> | 8 |
| 2.2.3 Hiper-parâmetros | 8 |
| 2.2.4 Redes Neurais | 9 |

| | | |
|----------|---|-----------|
| 2.2.5 | Validação Cruzada | 10 |
| 3 | Trabalho Relacionado | 13 |
| 4 | Modelo Proposto | 15 |
| 4.1 | Descrição Geral do Sistema | 15 |
| 4.2 | Requisitos | 15 |
| 4.3 | Abordagem | 17 |
| 4.3.1 | Construção do <i>Dataset</i> | 17 |
| 4.3.2 | Construção do Classificador | 20 |
| 4.3.3 | Desenvolvimento da Aplicação <i>web</i> | 21 |
| 5 | Implementação do Modelo | 23 |
| 5.1 | Construção do <i>Dataset</i> | 23 |
| 5.2 | Construção do Classificador | 24 |
| 5.3 | Desenvolvimento da Aplicação <i>web</i> | 25 |
| 6 | Validação e Testes | 27 |
| 7 | Conclusões e Trabalho Futuro | 29 |
| A | Um Detalhe Adicional | 31 |
| B | Outro Detalhe Adicional | 33 |
| | Bibliografia | 35 |

Lista de Tabelas

| | | |
|-----|--|----|
| 4.1 | Requisitos funcionais do sistema em desenvolvimento para a construção do <i>dataset</i> | 16 |
| 4.2 | Requisitos funcionais respeitantes ao processo de classificação. | 16 |
| 4.3 | Requisitos não funcionais do sistema. | 16 |
| 4.4 | Valores de duração (em segundos), amostras delizadas e N a considerar, para um valor de <i>samplingRate</i> igual a 44100Hz. . | 18 |

Lista de Figuras

| | | |
|-----|---|----|
| 2.1 | Características (cima e meio) e áudio (baixo) correspondente. . | 7 |
| 2.2 | Representação típica de uma rede neuronal. | 9 |
| 4.1 | Sistema de anotação de eventos desejado. | 15 |
| 4.2 | Matriz de características. | 18 |
| 4.3 | Vetor de classes. | 19 |
| 4.4 | Matriz de características. | 20 |
| 4.5 | <i>Design</i> desejado da página <i>web</i> 2. | 21 |
| 5.1 | Processo detalhado de construção do <i>dataset</i> | 24 |
| 5.2 | Resultados da aplicação de vários algoritmos aos dados. | 25 |
| 5.3 | Processo de construção do classificador. | 25 |
| B.1 | Descrição pormenorizada do campo de padel. [Tennisnerd, 2015] | 33 |

Capítulo 1

Introdução

A atividade desportiva de um atleta, numa determinada modalidade, deve ser monitorizada de uma perspetiva externa, no sentido de produzir melhores resultados. Para esse efeito, é necessário que o atleta seja supervisionado durante a sua atividade física, ou que a última seja registada e posteriormente analisada (registo em papel ou em vídeo).

Neste trabalho, a anotação de eventos sonoros em vídeo refere-se à análise do áudio extraído de um vídeo previamente gravado, com o objetivo de recolher informações relativas a possíveis batidas de bola. O processamento sobre o áudio será realizado em modo *offline*.

Este trabalho destina-se à análise da componente desportiva de padel ([Courel-Ibáñez et al., 2019]), no sentido de fornecer a atletas e/ou treinadores um complemento no decurso do treino.

Uma figura referente ao campo onde este desporto é praticado e à raquete e bola utilizadas encontra-se disponível no Capítulo A, na qual se observam as dimensões do campo onde esta modalidade desportiva é realizada. O desporto de padel pode ser praticado em ambiente interior (*indoor*) ou exterior (*outdoor*). No entanto, este trabalho analisa apenas vídeos em ambiente *indoor*.

1.1 Motivação

A aplicação em desenvolvimento implica que se implemente um sistema onde a máquina é capaz de reconhecer batidas de bola. No entanto, a forma como uma máquina reconhece um som difere de como esse mesmo som é percecio-

nado pelo ouvido humano. O ouvido humano consegue fazer a distinção entre sons, considerando as características dos sons emitidos: duração, intensidade, tom, entre outras [Council et al., 2004]. Pelo que, é necessário proporcionar à máquina a capacidade de reconhecer batidas de bola de forma semelhante à humana. Para esse efeito, será utilizada a automática.

A aprendizagem automática permite aplicar algoritmos aos dados recebidos pela máquina, e reconhecer nestes as batidas de bola.

Antes do processo de aprendizagem, não existindo dados relativos aos eventos pretendidos, ou que estejam etiquetados nas condições desejadas, será construído um *dataset* de raiz. Os dados constituintes do *dataset* serão obtidos de vídeos amadores.

A aplicação em desenvolvimento disponibiliza ainda uma interface, que permite que o utilizador participe no processo de anotação dos eventos (validando os resultados devolvidos pelo classificador), o que poderá contribuir numa perspetiva futura, para aumentar o *dataset* e melhorar deteção de batidas de bola.

1.2 Principais Contributos

A realização deste trabalho terá os seguintes contributos:

- Disponibilizar de um *dataset* construído de raiz;
- Providenciar de um modelo que reconhece batidas de bola;
- Proporcionar de uma aplicação *web* que permite aumentar o *dataset*.

1.3 Validação e Testes

(A alterar depois de fazer o capítulo de validação e testes)

1.4 Estrutura do Relatório

O relatório está organizado da seguinte forma:

- O Capítulo 1 introduz a necessidade do uso da aprendizagem automática para reconhecer batidas de bola em vídeos de treinos de padel, enuncia

os principais contributos da aplicação em desenvolvimento, e realiza uma apreciação global dos resultados obtidos.

- O Capítulo 2 faz referência a trabalhos relacionados com o trabalho corrente. Estes trabalhos foram sendo analisados durante o desenvolvimento do trabalho e baseiam-se também no uso de técnicas de aprendizagem automática para reconhecer eventos em modalidades desportivas.
- O Capítulo 3 introduz alguns conceitos nos quais o trabalho em desenvolvimento se baseia: conceitos sobre processamento de sinal e de aprendizagem automática.
- O Capítulo 4 visa apresentar uma análise geral dos requisitos do sistema e a abordagem utilizada para satisfazer esses requisitos.
- O Capítulo 5 descreve todo o processo de concretização da solução abordada no capítulo anterior.
- (Falar do capítulo 6)
- (Falar do capítulo 7)

Capítulo 2

Fundamentos

Neste capítulo serão abordados os conceitos de carácter teórico, que sustentam o trabalho realizado.

2.1 Conceitos de Processamento de Sinal

Um sinal corresponde a uma grandeza física que varia ao longo do tempo. Em termos matemáticos, um sinal varia em função de uma ou mais variáveis, que podem ser contínuas (sinais contínuos) ou discretas (sinais discretos) [Sampaio et al., 2006].

Neste trabalho, será feita uma análise sobre áudio que corresponde a um sinal discreto no tempo.

2.1.1 Amostragem

O processo de amostragem converte um sinal contínuo num sinal discreto. Para isso, escolhe pontos equi-espaçados do sinal contínuo para gerar os pontos do sinal discreto [LibreTexts, 2022a]. Cada um dos pontos escolhidos é designado por amostra.

O intervalo entre as amostras consecutivas denomina-se período de amostragem e é dado em segundos(s). O inverso deste valor corresponde à frequência de amostragem, que é dada em *Hertz*(Hz), e refere-se ao número de amostras que ocorre no sinal a cada segundo. A informação removida pelo processo de amostragem pode ser parcialmente recuperada através de interpolação ou reconstrução do sinal [LibreTexts, 2022b].

O ouvido humano deteta sons na gama de frequências compreendida entre 20Hz e 20kHz, pelo que, de acordo com o teorema de *Nyquist* [Evia e Arnold, 2022], é necessário que a frequência de amostragem dos sons que chegam ao aparelho auditivo humano sejam amostrados a uma frequência duas vezes igual ou superior a 20kHz.

A frequência de amostragem a ser considerada neste projeto é de 44100Hz (cerca de 43kHz), que é um dos valores padrão utilizados [Brown, 2019].

2.1.2 Transformada de Fourier

A transformada de Fourier (*Fourier Transform* – FT) é uma função matemática que decompõe um sinal nas suas várias frequências (sinusóides) constituintes [Semmlow, 2012]. O sinal de entrada pertence ao domínio do tempo, e o sinal de saída pertence ao domínio da frequência. Esta função (nos domínios do tempo e frequência) é designada por espectro.

A transformada de *Fourier* é útil para verificar, por exemplo, em que zonas do espectro existe mais ou menos energia concentrada e será utilizada no cálculo das características a extrair do áudio.

2.1.3 Características extraídas do áudio

Neste trabalho, a partir do áudio são extraídas as seguintes características: deteção do início de um som (*Onset*) e o valor eficaz (RMS).

O *Onset* divide-se em duas componentes: a deteção do início (*Onset Detect*) e o fluxo espectral (*Spectral Flux*). A primeira componente, corresponde a detetar o instante em que se inicia um determinado som [Rosão, 2012]. A segunda corresponde a verificar variações no espectro do sinal, no sentido de encontrar diferenças entre *frames* consecutivas, o que permite detetar também o início de um som. O fluxo espectral pode também ser definido como uma medida do quão depressa o espectro de um sinal varia [Meyda, 2022].

O valor eficaz (*Root Mean Square* – RMS) refere-se à energia (intensidade) média concentrada numa *frame* [Room, 2021].

A figura 2.1 constitui uma representação das três características enunciadas.

Observando a figura, verifica-se que os picos das duas componentes do *Onset*, bem como o RMS estão correspondem aos instantes no áudio original

onde a amplitude é maior.

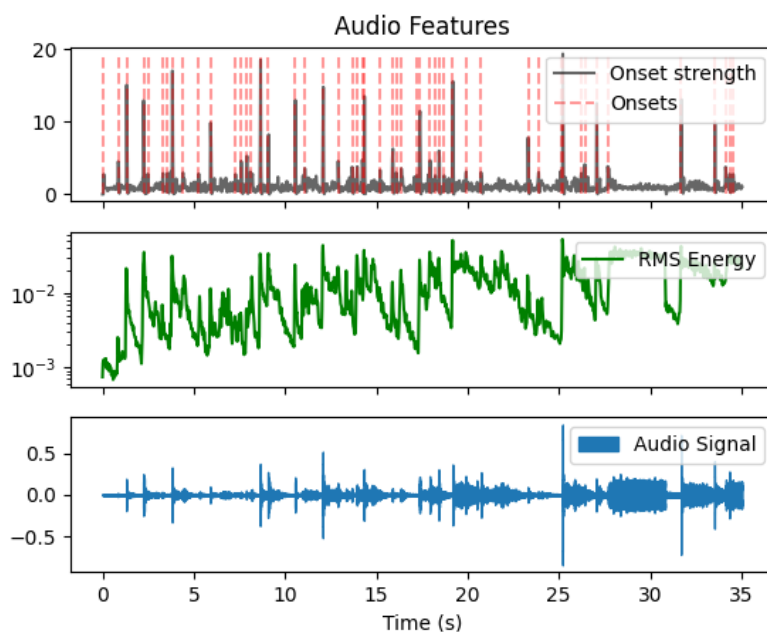


Figura 2.1: Características (cima e meio) e áudio (baixo) correspondente.

As características enunciadas acima podem ser calculadas com o auxílio da biblioteca *librosa* [McFee et al., 2015]. Este módulo realiza processamento sobre sinais de áudio, no sentido de recolher informações sobre o mesmo.

2.2 Conceitos de Aprendizagem Automática

A aprendizagem automática (*Machine Learning* - ML) é ramo da inteligência artificial que recorre à análise de grandes volumes de dados visando extrair padrões e representá-los num modelo [Janiesch et al., 2021].

Existem três tipos de aprendizagem automática [Sah, 2020]:

- Supervisionada – O modelo é construído com base num conjunto de dados (*dataset*) que associa uma lista de características (*features*) à classe correspondente;
- Não Supervisionada – O modelo é construído com base num conjunto de dados (*dataset*) que apenas contém as características;

- Por Reforço – O modelo é contruído com base na interação com um sistema que gera os dados necessários à extração de padrões.

Este trabalho utiliza aprendizagem supervisionada para extrair os padrões de áudio (e.g., batida de bola em raquete).

2.2.1 *Dataset*

Um *dataset* é um conjunto de dados usado no processo de construção dos modelos gerados através de métodos de aprendizagem automática. O *dataset* é estruturado em colunas e linhas (formato de tabela). No caso da aprendizagem supervisionada as colunas representam as características (*features*) e a classe. As linhas representam sempre instâncias (exemplos ou observações).

Considerando uma matriz de características, X , e o vetor de valores de classes correspondente, y , o problema de classificação em questão tem como objetivo obter a função, f , tal que:

$$f(X) = y \quad (2.1)$$

A função f corresponde a um modelo que deve receber a matriz de características e, classificar de maneira correta cada um dos exemplos contituíntes.

2.2.2 *Desiquilíbrio no Dataset*

O desequilíbrio no *dataset* ocorre quando a distribuição de exemplos pelas classes é desigual [Badr, 2019]. Esse desequilíbrio pode resultar na construção de um modelo enviesado que reconhece melhor determinadas classes.

Num problema de classificação binária o desequilíbrio pode abordar-se aumentando o número de exemplos da classe em minoria (sobre-amostragem) ou reduzindo o número da classe em maioria (sub-amostragem) [Badr, 2019].

2.2.3 *Hiper-parâmetros*

Em aprendizagem automática, um hiper-parâmetro é um valor utilizado pelo modelo para controlar o processo de treino. Os hiper-parâmetros de um modelo não são utilizados diretamente no modelo, mas determinam os parâmetros internos utilizados diretamente [Nyuytiymbiy, 2021]. O prefixo

“hiper” advém do facto de estes parâmetros serem os escolhidos para definir outros, sugerindo que estão num nível hierárquico superior.

2.2.4 Redes Neurais

Uma rede neuronal artificial (*Artificial Neural Network* – ANN) é uma técnica de aprendizagem automática que tem como inspiração a biologia humana e a forma como os neurónios comunicam entre si para compreender os dados percecionados [Rauber, 4].

Uma rede neuronal está organizada em camadas, que são constituídas por unidades (nós). A primeira camada (*deinput*), tem um número de nós igual ao número de características do *dataset*. A última camada (*de output*), possui um número de nós igual ao número de classes. Entre a camada de *input* e a camada de *output* existem uma ou mais camadas intermédias. A figura 2.2 representa de uma rede neuronal, onde estão delimitadas as ligações entre os vários nós constituintes:

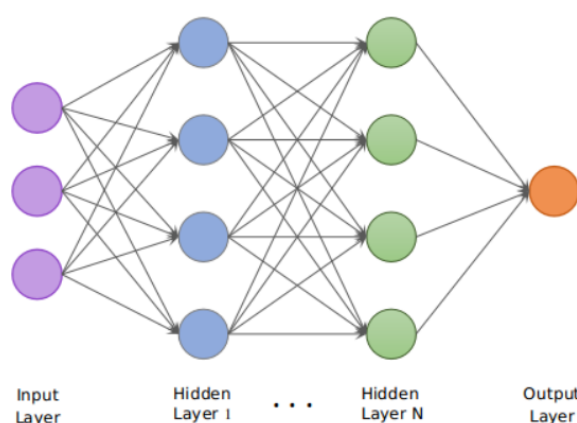


Figura 2.2: Representação típica de uma rede neuronal.

O número de camadas, bem como o número de nós em cada camada são definidos com o objetivo de obter o melhor classificador possível [Krishnan, 2021].

Tal como se verifica na figura 2.2, cada nó recebe os valores de saída dos nós da camada anterior e associa-lhes um peso que é atualizado ao longo do processo de aprendizagem [Medium, 2019].

O processo de aprendizagem é realizado ao longo de várias épocas. Uma época (*epoch*) corresponde a um ciclo de passagem, pela rede neuronal, de

todos os exemplos do *dataset*. Uma época é constituída por várias iterações. Cada iteração corresponde a uma atualização de pesos após a passagem pela rede de um número pré-definido (*batch*) de exemplos.

2.2.5 Validação Cruzada

No caso da aprendizagem supervisionada, a classificação é utilizada para atribuir uma classe (categoria) a um conjunto de exemplos. Após este processo, é construído um modelo que através dos padrões presentes nos dados de *input*, prevê a classe à qual esses dados pertencem. Existem três tipos de classificação [Brownlee, 2022]:

- Multi-Classe – quando existem mais do que duas classes nos dados. Um exemplo não pode pertencer, simultaneamente, a mais do que uma classe.
- Binária – quando existem apenas duas classes nos dados. Um exemplo não pode pertencer, simultaneamente, a mais do que uma classe.
- Multi-Label – quando existem duas ou mais classes e um exemplo pode pertencer, simultaneamente, a mais do que uma classe.

Neste trabalho, o problema de classificação é binário, uma vez que o modelo deve distinguir exemplos que são batidas de bola (positivos) de exemplos que não são batidas de bola (negativos).

Para avaliar o desempenho de um classificador é necessário verificar quais são as hipóteses de este acertar independentemente da classe, mas também verificar a distribuição de acertos em cada classe. Essas probabilidades podem ser obtidas através da matriz de confusão. Neste trabalho, entre as várias métricas de desempenho que podem ser obtidas da matriz de confusão, serão consideradas as seguintes [Brownlee, 2020]:

- Taxa de sucesso (*Accuracy*) – percentagem de exemplos bem classificados independentemente da classe;
- Precisão (*Precision*) – percentagem de exemplos que o classificador indicou serem de uma determinada classe e que pertencem a essa classe;

- Cobertura (*Recall*) – percentagem de exemplos que pertencem a uma determinada classe e o classificador indicou como sendo dessa classe;
- *F1-score* – é a média harmónica entre a precisão e a cobertura.

Em geral, num problema de classificação os dados são divididos em treino e teste. O grupo de exemplos de treino é utilizado no processo de aprendizagem, e o grupo de teste é utilizado no processo de classificação.

Esta distribuição dos dados pode não ser suficiente para que o modelo aprenda a generalizar, ao analisar dados novos. Como solução, pode ser utilizada validação.

Um caso particular da validação é a técnica de validação cruzada [Alhamid, 2020]. Esta técnica divide os dados em K grupos (*folds*) e ao longo de K iterações treina modelos diferentes onde $K-1$ *folds* são utilizados para treino e um *fold* é utilizado para validação.

Neste trabalho, será utilizada a técnica de validação cruzada estratificada (*Stratified K-Fold*) que permite manter a distribuição dos exemplos de cada classe nos *folds*.

Capítulo 3

Trabalho Relacionado

(Por fazer)

Capítulo 4

Modelo Proposto

Neste capítulo será realizada uma descrição geral do sistema desenvolvido, dos requisitos e da abordagem adotada para o desenvolver.

4.1 Descrição Geral do Sistema

O funcionamento do sistema pode ser representado como se mostra na figura 4.1. O sistema desenvolvido reconhece batidas de bola, permite auxiliar no processo de construção do dataset e, conseqüentemente no processo de melhoria de desempenho do classificador.

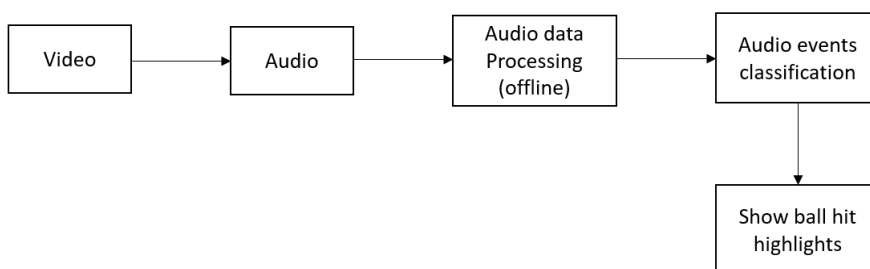


Figura 4.1: Sistema de anotação de eventos desejado.

4.2 Requisitos

Os requisitos do sistema podem ser funcionais (funções de sistema) e não funcionais (atributos do sistema). As funções de sistema, referem-se a funciona-

lidades desempenhadas pelo sistema. Neste projeto, os requisitos funcionais foram organizados em dois grupos:

- Construção do *dataset*;
- Construção do classificador.

A tabela 4.1 contém os requisitos funcionais que dizem respeito à criação do *dataset*:

| Requisito | Função | Categoria |
|-----------|--|-----------|
| R1.1 | Extrair o áudio do vídeo | Invisível |
| R1.2 | Anotar as batidas de bola presentes no áudio (A) | Invisível |
| R1.3 | Extrair as características do áudio (B) | Invisível |
| R1.4 | Combinação dos dois conjuntos A e B | Invisível |

Tabela 4.1: Requisitos funcionais do sistema em desenvolvimento para a construção do *dataset*.

A tabela 4.2 contém os requisitos funcionais que dizem respeito ao processo de classificação:

| Requisito | Função | Categoria |
|-----------|---|-----------|
| R1.1 | Extrair o áudio do vídeo | Invisível |
| R1.2 | Extrair as características do áudio | Invisível |
| R1.3 | Aplicar o classificador | Invisível |
| R1.4 | Apresentar resultados dos eventos etiquetados | Invisível |

Tabela 4.2: Requisitos funcionais respeitantes ao processo de classificação.

Os atributos do sistema descrevem a forma como o sistema deve ser disponibilizado. A tabela 4.3 representa os atributos do sistema.

| Atributo | Detalhe/ Restrição Fronteira | Categoria |
|-------------------------|---|-------------|
| Interação Homem-Máquina | Detalhe Interface fácil de aprender a usar | Obrigatório |

Tabela 4.3: Requisitos não funcionais do sistema.

4.3 Abordagem

Neste secção será descrita a abordagem utilizada para realizar cada uma das etapas inerentes ao sistema em desenvolvimento. As etapas são as seguintes:

- Construção do *dataset*;
- Construção de um classificador que reconhece batidas de bola;
- Desenvolvimento de uma aplicação *web* que contribui para aumentar o *dataset*.

4.3.1 Construção do *Dataset*

O *dataset* construído, corresponde a um conjunto de dados com padrões identificativos de batidas de bola.

Obtenção da matriz de características – X

No processo de extração das características (enunciadas na secção 2.1.3) é realizado um varrimento sobre o áudio com uma janela fixa. Este processo é realizado ao longo de várias iterações, onde em cada iteração a janela desliza um número específico de amostras, a partir das quais são extraídas as características pretendidas.

No cálculo dessas características, o *package librosa* considera um salto (“*hop*”) de 512 amostras. Pelo que, este módulo divide o sinal em segmentos, com 512 amostras de dimensão, e sobre cada um deles calcula o valor da característica [McFee et al., 2015]. No entanto, podem ser escolhidos múltiplos do valor do salto como outras abordagens, no sentido de verificar que alterações produzem no desempenho do classificador. O mesmo se aplica ao tamanho da janela: uma batida de bola dura em média meio segundo, mas podem ser realizados experimentos com durações maiores (na tabela 4.4). Neste trabalho, como primeira abordagem, será considerada uma janela de 0.5 segundos e um salto de 512 amostras.

A matriz X é constituída por M linhas ou exemplos. Por sua vez, cada exemplo tem N colunas ou valores de características, pelo que a matriz X pode ser representada na figura 4.2 da seguinte forma:

$$X = \begin{bmatrix} X_1 & X_2 & X_3 & \cdots & X_N \\ X_2 & X_3 & X_4 & \cdots & X_{N+1} \\ X_3 & X_4 & X_5 & \cdots & X_{N+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ X_{N-1} & X_N & X_{N+1} & \cdots & X_{2N-2} \\ X_N & X_{N+1} & X_{N+2} & \cdots & X_{2N-1} \\ X_{N+1} & X_{N+2} & X_{N+3} & \cdots & X_{2N} \\ \vdots & \vdots & \vdots & & \vdots \end{bmatrix}$$

Figura 4.2: Matriz de características.

O valor de N na figura 4.2, corresponde ao número de segmentos de amostras abranjidos pela janela fixa para uma característica. O valor de N pode ser dado pela seguinte expressão:

$$N = \frac{eventLength \times samplingRate}{hopLength} \quad (4.1)$$

onde na equação, *eventLength* corresponde à duração de um evento ou batida de bola, *samplingRate* é a frequência de amostragem (a que os áudios são obtidos), e *hopLength* refere-se ao número de amostras deslizadas a cada iteração do varrimento. Na tabela 4.4, estão evidenciados os valores de N obtidos para as várias combinações de *eventLength* e *hopLength*:

| <i>eventLength</i> | <i>hopLength</i> | <i>N</i> |
|--------------------|------------------|----------|
| 0.5 | 512 | 43 |
| 0.5 | 1024 | 22 |
| 0.5 | 2048 | 11 |
| 0.7 | 512 | 60 |
| 0.7 | 1024 | 30 |
| 0.7 | 2048 | 15 |
| 1.0 | 512 | 86 |
| 1.0 | 1024 | 43 |
| 1.0 | 2048 | 22 |

Tabela 4.4: Valores de duração (em segundos), amostras delizadas e N a considerar, para um valor de *samplingRate* igual a 44100Hz.

Voltando ao processo de deslizamento da janela e observando ainda a

figura 4.2, a primeira iteração permite obter a primeira linha da matriz X . A segunda iteração permite obter a segunda linha da matriz e, assim sucessivamente até à iteração M . Note-se que o valor de N , corresponde ao número de segmentos de amostras obtidos para uma característica. Por isso, caso se pretenda considerar as três características enunciadas na secção 2.1.3, o número de elementos em cada linha da matriz X , é igual a três vezes o valor de N .

Obtenção do vetor de classes – y

Cada um dos exemplos da matriz de características pertence a uma classe ou categoria. O vetor de classes pode ser representado da seguinte forma:

$$y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_M \end{bmatrix}$$

Figura 4.3: Vetor de classes.

onde cada elemento y_i corresponde à classe etiquetada. Para obtenção deste vetor, é necessário etiquetar manualmente o áudio. O processo resume-se a ouvir o áudio e anotar num ficheiro .csv, em que intervalos de tempo ocorrem batidas de bolas.

Note-se que apenas as batidas de bola são registadas no ficheiro. Por isso, todos os eventos que não são registados constituem ruído. Cada um dos exemplos, y_i , do vetor y terá um valor dependente da expressão 4.2:

$$y_i = \begin{cases} \text{Ball hit} & \forall X_i: X_i \in [\text{beginSample}, \text{endSample}] \\ \text{Non ball hit} & \forall X_i: X_i \notin [\text{beginSample}, \text{endSample}] \end{cases} \quad (4.2)$$

onde *beginSample* é a amostra inicial do evento e *endSample* corresponde à amostra final constituinte desse mesmo evento.

Criação do *dataset*

O processo de construção do *dataset* refere-se à junção da matriz de características ao vetor de classes. A janela desliza ao longo do áudio por isso, um conjunto de exemplos ou instâncias, X_i , corresponderão a uma mesma classe, y_i , tal como se verifica na figura 4.4:

$$X = \begin{bmatrix} X_1 & X_2 & X_3 & \cdots & X_N & y_1 \\ X_2 & X_3 & X_4 & \cdots & X_{N+1} & y_1 \\ X_3 & X_4 & X_5 & \cdots & X_{N+2} & y_1 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \\ X_{N-1} & X_N & X_{N+1} & \cdots & X_{2N-2} & y_1 \\ X_N & X_{N+1} & X_{N+2} & \cdots & X_{2N-1} & y_1 \\ X_{N+1} & X_{N+2} & X_{N+3} & \cdots & X_{2N} & y_1 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \end{bmatrix}$$

Figura 4.4: Matriz de características.

As linhas ou instâncias seguintes, que não estão evidenciadas na matriz, estão associadas a outra classe e, assim sucessivamente.

O vídeo de *input*, a partir do qual o *dataset* foi construído é um vídeo obtido de forma não profissional e tem a duração de 1 hora e 32 minutos. O vídeo em questão constitui uma gravação de um treino realizado por um conjunto de atletas em ambiente *indoor*.

4.3.2 Construção do Classificador

Antes de construir o classificador, foi necessário escolher o algoritmo de aprendizagem automática a utilizar. Para esse efeito, fez-se uso da ferramenta *Orange Data Mining* – ODM que permite simular a aplicação de vários algoritmos aos dados, no sentido de verificar quais destes produzem os melhores resultados [Baptista, 2019].

A construção do classificador envolve também a escolha dos hiper-parâmetros que produzem os melhores resultados.

4.3.3 Desenvolvimento da Aplicação *web*

A aplicação *web* desenvolvida tem duas páginas: página (1) que permite escolher o vídeo sobre o qual se pretende visualizar os resultados; página (2) que permite visualizar os resultados devolvidos pelo classificador (para um determinado vídeo). O *design* destas páginas encontra-se nas figuras ?? e 4.5. Todo o processamento efetuado sobre o áudio é realizado antes de mostrar os resultados nesta aplicação.

(Colocar aqui o design da página 1).

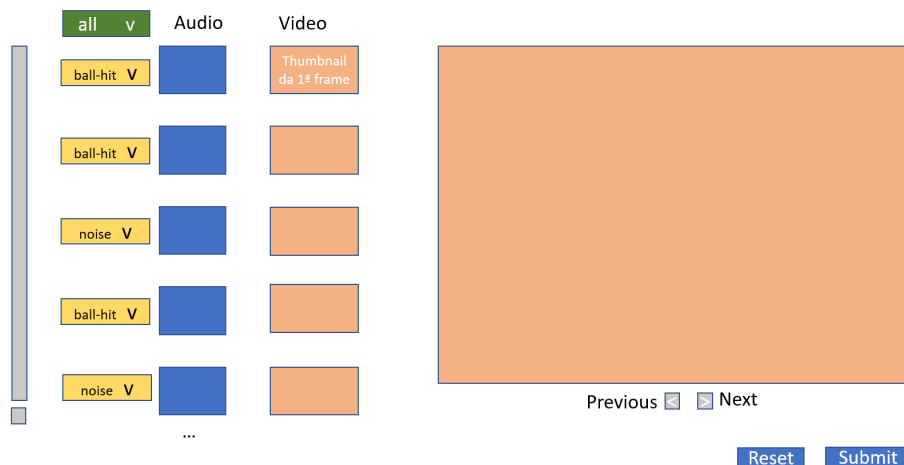


Figura 4.5: *Design* desejado da página *web* 2.

Capítulo 5

Implementação do Modelo

Este capítulo descreve todo o processo de implementação das etapas enunciadas na secção abordagem (4.3).

5.1 Construção do *Dataset*

O processo de criação do *dataset* pode ser representado pelo esquema que se encontra na figura 5.1.

À exceção da fase de anotação manual das batidas de bola (“*Human ear annotation*”), todas as outras fases do esquema foram realizadas no ambiente de desenvolvimento *Pycharm* que compila e executa código *Python* [Jetbrains, 2022].

Para simplificar o processo de anotação das batidas de bola, optou-se por segmentar o vídeo de longa duração em vídeos com 1 minuto. O processo de segmentação deu origem a cerca de 90 vídeos de curta duração. Para cada um desses vídeos, foram gerados os áudios correspondentes. De todos os áudios gerados anotaram-se 24.

A anotação das batidas de bola foi realizada com o auxílio da aplicação *Adobe Audition* [Adobe, 2013]. Esta ferramenta permite efetuar uma análise sobre áudio e verificar em que instantes ocorrem batidas de bola. A anotação é realizada num ficheiro *Excel*, que posteriormente é convertido para *CSV*.

Cada evento está registado no ficheiro *Excel* com seguintes dados:

- O tipo de batida de bola (em raquete, no chão, na rede metálica, entre outros). Todos estes tipos pertencem à classe batida de bola;

- O intervalo de tempo em que ocorre a batida de bola (amostras inicial e final no áudio);
- O tipo de ambiente em que o treino é praticado (*indoor* ou *outdoor*);
- O lado do campo em que batida de bola ocorre.

O tipo de ambiente onde o desporto é praticado e o lado do campo onde ocorrem as batidas de bola serão apenas analisados em trabalho futuro.

Na fase de extração de características iterou-se apenas sobre cada um dos áudios anotados. Sobre estes áudios realizou-se o processo descrito na secção 4.3.1 para obtenção da matriz de características.

Na fase de “*Labeling*” combinou-se a informação registada nos ficheiros *CSV* (batidas de bola) com as características tal como se verifica na figura 4.4, o que permitiu obter o *dataset* a utilizar na construção do modelo (“*Input data*”). Este processo de etiquetação foi realizado com base na equação 4.1.

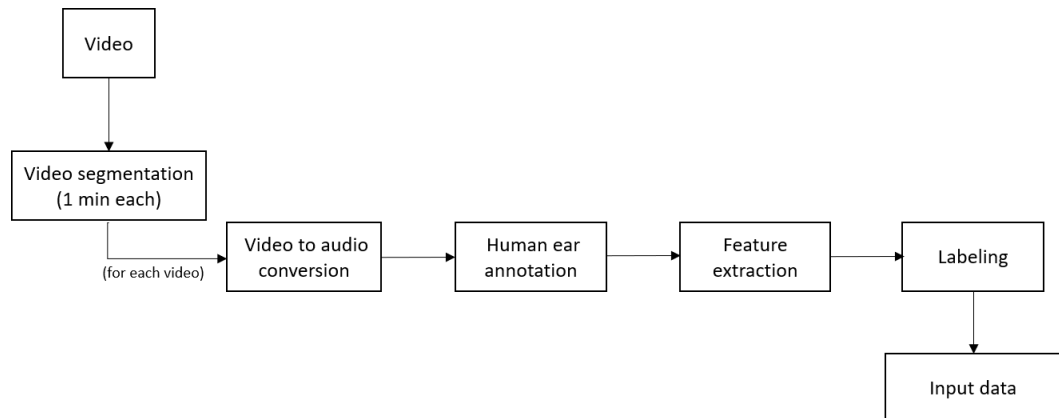


Figura 5.1: Processo detalhado de construção do *dataset*.

Considerando como primeira abordagem a primeira linha da tabela 4.4, o *dataset* obtido contém cerca de 130 mil exemplos.

5.2 Construção do Classificador

A construção do modelo pode ser descrita através da figura ??.

Através da ferramenta *Orange Data Mining* obtiveram-se os resultados que se encontram na figura 5.2.

Esta figura contém os resultados referentes à aplicação de vários algoritmos ao *dataset*.

Para verificar a qualidade dos modelos é utilizada validação cruzada estratificada (“*Stratified*”). Neste processo de validação os dados são divididos em 5 *folds*. A figura 5.2 sugere que os modelos que produzem os melhores resultados são o regressor logístico e a rede neuronal. O regressor logístico pode ser “visto” como uma rede neuronal com apenas uma camada intermédia [DataScienceCentral, 2019]. Pelo que neste trabalho será utilizada a rede neuronal como algoritmo de aprendizagem automática.

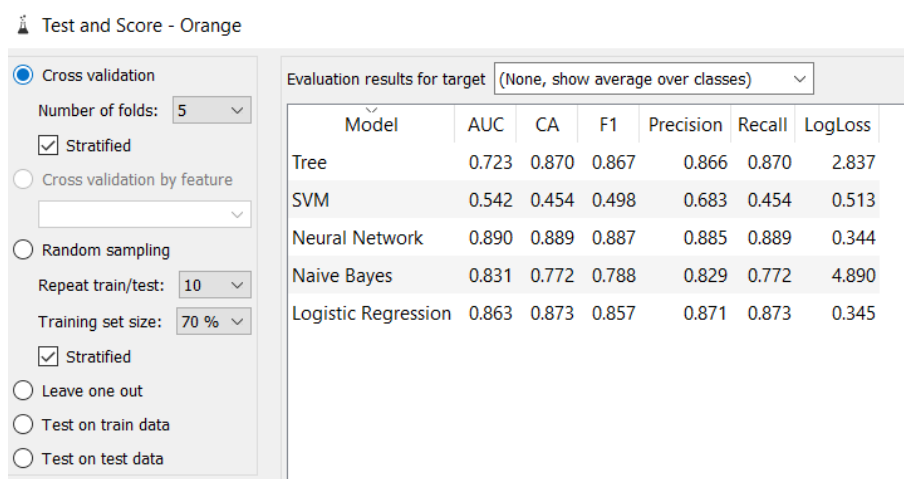


Figura 5.2: Resultados da aplicação de vários algoritmos aos dados.

(Terminar racicínio relativo ao esquema da figura 5.3)

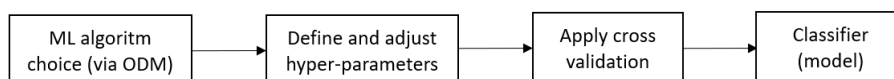


Figura 5.3: Processo de construção do classificador.

5.3 Desenvolvimento da Aplicação *web*

(Por fazer)

Capítulo 6

Validação e Testes

(Por fazer)

Capítulo 7

Conclusões e Trabalho Futuro

(Por fazer)

Apêndice A

Um Detalhe Adicional

O “apêndice” utiliza-se para descrever aspectos que tendo sido desenvolvidos pelo autor constituem um complemento ao que já foi apresentado no corpo principal do documento.

Neste documento utilize o apêndice para explicar o processo usado na **gestão das versões** que foram sendo construídas ao longo do desenvolvimento do trabalho.

É especialmente importante explicar o objetivo de cada ramo (“branch”) definido no projeto (ou apenas dos ramos mais importantes) e indicar quais os ramos que participaram numa junção (“merge”).

É também importante explicar qual a arquitetura usada para interligar os vários repositórios (e.g., Git, GitHub, DropBox, GoogleDrive) que contêm as várias versões (e respetivos ramos) do projeto.

Notar a diferença essencial entre “apêndice” e “anexo”. O “apêndice” é um texto (ou documento) que descreve trabalho desenvolvido pelo autor (e.g., do relatório, monografia, tese). O “anexo” é um texto (ou documento) sobre trabalho que não foi desenvolvido pelo autor.

Para simplificar vamos apenas considerar a noção de “apêndice”. No entanto, pode sempre adicionar os anexos que entender como adequados.

Apêndice B

Outro Detalhe Adicional

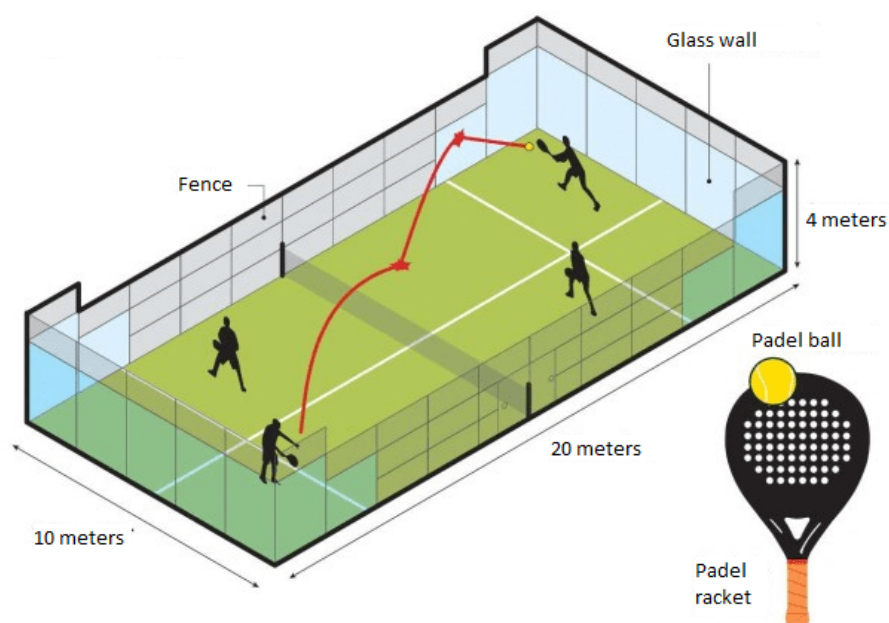


Figura B.1: Descrição pormenorizada do campo de padel. [Tennisnerd, 2015]

Bibliografia

- [Adobe, 2013] Adobe, U. (2013). Adobe audition.
- [Alhamid, 2020] Alhamid, M. (2020). What is cross-validation? <https://towardsdatascience.com/what-is-cross-validation-60c01f9d9e75>.
- [Badr, 2019] Badr, W. (2019). Having an imbalanced dataset? here is how you can fix it. *Towards Data Science*, 22.
- [Baptista, 2019] Baptista, B. (2019). Machine learning sem código. <https://medium.com/ensina-ai/machine-learning-sem-c%C3%B3digo-636d1a8f9081>.
- [Brown, 2019] Brown, G. (2019). Digital audio basics: sample rate and bit depth. *Dostupno na: https://www.izotope.com/en/learn/digital-audio-basics-sample-rate-and-bit-depth.html [15. rujna 2020.]*.
- [Brownlee, 2020] Brownlee, J. (2020). Tour of evaluation metrics for imbalanced classification. *Vermont Victoria*.
- [Brownlee, 2022] Brownlee, J. (2022). Types of classification tasks in machine learning. *Machine Learning Mastery*, 4p. Available online at: <https://machinelearningmastery.com/types-of-classification-in-machine-learning> (accessed November 25, 2021).
- [Council et al., 2004] Council, N. R. et al. (2004). Hearing loss: Determining eligibility for social security benefits.
- [Courel-Ibáñez et al., 2019] Courel-Ibáñez, J., Martinez, B. J. S.-A., e Marín, D. M. (2019). Exploring game dynamics in padel: Implications for assessment and training. *The Journal of Strength & Conditioning Research*, 33(7):1971–1977.

- [DataScienceCentral, 2019] DataScienceCentral (2019). Logistic regression as a neural network. <https://www.datasciencecentral.com/logistic-regression-as-a-neural-network/>.
- [Evia e Arnold, 2022] Evia e Arnold (visitado em Jul.2022). Nyquist sampling theorem.
- [Janiesch et al., 2021] Janiesch, C., Zschech, P., e Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets*, 31(3):685–695.
- [Jetbrains, 2022] JetBrains (2022). Pycharm. <https://www.jetbrains.com/pycharm/>.
- [Krishnan, 2021] Krishnan, S. (2021). How to determine the number of layers and neurons in the hidden layer. *URL: https://medium.com/geekculture/introduction-to-neuralnetwork-2f8b8221fbd3*.
- [LibreTexts, 2022a] LibreTexts, E. (2022a). Signal sampling. [https://eng.libretexts.org/Bookshelves/Electrical_Engineering/Signal_Processing_and_Modeling/Signals_and_Systems_\(Baraniuk_et_al.\)/10%3A_Sampling_and_Reconstruction/10.01%3A_Signal_Sampling](https://eng.libretexts.org/Bookshelves/Electrical_Engineering/Signal_Processing_and_Modeling/Signals_and_Systems_(Baraniuk_et_al.)/10%3A_Sampling_and_Reconstruction/10.01%3A_Signal_Sampling).
- [LibreTexts, 2022b] LibreTexts, E. (2022b). Signal sampling. [https://eng.libretexts.org/Bookshelves/Electrical_Engineering/Signal_Processing_and_Modeling/Signals_and_Systems_\(Baraniuk_et_al.\)/10%3A_Sampling_and_Reconstruction/10.03%3A_Signal_Reconstruction](https://eng.libretexts.org/Bookshelves/Electrical_Engineering/Signal_Processing_and_Modeling/Signals_and_Systems_(Baraniuk_et_al.)/10%3A_Sampling_and_Reconstruction/10.03%3A_Signal_Reconstruction).
- [McFee et al., 2015] McFee, B., Raffel, C., Liang, D., Ellis, D. P., McVicar, M., Battenberg, E., e Nieto, O. (2015). librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, volume 8, p. 18–25. Citeseer.
- [Medium, 2019] Medium (2019). A beginner intro to neural networks. <https://purnasaigudikandula.medium.com/a-beginner-intro-to-neural-networks-543267bda3c8>.
- [Meyda, 2022] Meyda (visitado em Jul.2022). Audio feature extraction for javascript. <https://meyda.js.org/audio-features.html>.

- [Nyuytiymbiy, 2021] Nyuytiymbiy, K. (2021). Parameters and hyperparameters in machine learning and deep learning. *Medium*. <https://towardsdatascience.com/parameters-and-hyperparameters-aa609601a9ac> (April 22, 2021).
- [Rauber, 4] Rauber, T. W. (4). Redes neuronais artificiais. *Documento de Apoio* (visitado em Jul, 2022).
- [Room, 2021] Room, C. (2021). Audio feature extraction. *machine learning*, 16(17):51.
- [Rosão, 2012] Rosão, C. M. T. (2012). *Onset detection in music signals*. PhD thesis.
- [Sah, 2020] Sah, S. (2020). Machine learning: a review of learning types.
- [Sampaio et al., 2006] Sampaio, R., Cataldo, E., e BRANDÃO, A. (2006). Análise e processamento de sinais. *Sociedade Brasileira de Matemática Aplicada e Computacional*. São Paulo.
- [Semmlow, 2012] Semmlow, J. (2012). Chapter 3 - fourier transform: Introduction. In Semmlow, J., editor, *Signals and Systems for Bioengineers (Second Edition)*, Biomedical Engineering, p. 81–129. Academic Press, second edition edition.
- [Tennisnerd, 2015] Tennisnerd (2015). What is padel tennis? <https://tennisnerd.net/padel/what-is-padel-tennis/15893>.