

**INSTITUTO SUPERIOR DE ENGENHARIA DE  
LISBOA**

Área Departamental de Engenharia de Electrónica e  
Telecomunicações e de Computadores

**Dissertação de Mestrado em Engenharia de Redes de  
Comunicação e Multimédia**

**Classificação de atividades físicas através do uso  
do acelerómetro do *Smartphone***

**FRANCISCO JOÃO AIRES DUARTE**

Dissertação de natureza científica realizada para a obtenção do grau de Mestre  
em Engenharia de Redes de Comunicação e Multimédia

Orientadores:

Doutor Arnaldo Abrantes

Mestre André Ribeiro Lourenço

Júri:

Presidente: Doutor Pedro Miguel Torres Mendes Jorge

Vogais:

Mestre Hugo Humberto Plácido da Silva

Doutor Arnaldo Abrantes

Mestre André Ribeiro Lourenço

**Outubro de 2013**





**INSTITUTO SUPERIOR DE ENGENHARIA DE  
LISBOA**

Área Departamental de Engenharia de Electrónica e  
Telecomunicações e de Computadores

**Mestrado em Engenharia de Redes de Comunicação e  
Multimédia**

**Classificação de atividades físicas através do uso  
do acelerómetro do *Smartphone***

**FRANCISCO DUARTE**

Dissertação de natureza científica realizada para a obtenção do grau de Mestre  
em Engenharia de Redes de Comunicação e Multimédia

Orientadores:

Doutor Arnaldo Abrantes

Mestre André Ribeiro Lourenço

Júri:

Presidente: Doutor Pedro Miguel Torres Mendes Jorge

Vogais:

Mestre Hugo Humberto Plácido da Silva

Doutor Arnaldo Abrantes

Mestre André Ribeiro Lourenço

**Outubro de 2013**



*"The best touch interface is sometimes no touch at all.*

*Using sensors and speech is the next frontier for interaction design."*

Josh Clark

# *Resumo*

Os *smartphones* desempenham hoje um papel ubíquo no acesso e processamento de informação, sendo que, na sua grande maioria, possuem uma miríade de sensores integrados que os torna capazes de gerar informação de elevada precisão e rigor. A monitorização do exercício físico apresenta-se como uma das novas tendências possibilitada pela utilização destes dispositivos. Sensores de movimento, como o acelerómetro, possibilitam a caracterização de movimento. Este trabalho pretende estudar esta temática e desenvolver uma aplicação para o sistema operativo Android, que tira partido dos sensores embutidos nos *smartphones* e das tecnologias web, com o objetivo de realizar classificação de atividades.

A solução desenvolvida é feita com base na arquitetura cliente-servidor. A aplicação cliente realiza a aquisição de dados, visualização e gravação do sinal obtido através do acelerómetro do *smartphone* e a aplicação servidor recebe a informação adquirida pelo cliente, processa-a e classifica-a.

De modo a ser possível classificar movimentos em atividades foi realizada uma extensa análise dos sinais adquiridos por forma a perceber quais as características mais distintivas e formulou-se o problema em análise do paradigma da aprendizagem supervisionada tendo sido estudados vários tipos de classificadores. O estudo de utilização já realizado, usou um cenário típico de utilização, posicionando o *smartphone* junto à cintura dentro do bolso frontal direito, na tentativa de simular condições o mais naturais possíveis.

A exploração preliminar da extração de características tanto no domínio da frequência como no domínio do tempo e classificadores paramétricos e não paramétricos permitiu obter resultados preliminares que demonstram que a classificação de atividades é realizada com elevada precisão ( $> 95\%$ ).

**Palavras-chave:** Acelerómetro, Android, *Smartphone*, Classificação de atividade, *Quantified-Self*

# *Abstract*

Nowadays, smartphones play an ubiquitous role in accessing and processing information, most of them having a myriad of integrated sensors that makes them capable of generating information with high accuracy and precision. The monitoring of physical exercises presents itself as one of the new trends, made possible by the use of devices like smartphones. Motion sensors such as the accelerometer enable live motion measurement. This paper intends to study this issue and develop an application for the Android operating system, which takes advantage of sensors embedded in smartphones and web technologies, with the goal to classify multiple physical activities.

The developed solution is based on client-server architecture. The client application performs data acquisition, visualization and recording of the signal obtained by the smartphone's accelerometer and the server application receives the information acquired by the client, processes it and classifies it.

In order for the application to be able to classify multiple movements throughout the activity performed by the user, an extensive analysis of the acquired signals was carried out to understand its most distinctive features. We used a supervised approach with the goal of reviewing the best techniques that should be useful for achieving the classification with the lowest error. For the signals acquisition the smartphone was positioned along the waist, inside the right front pocket in an attempt to simulate conditions as natural as possible.

The study explored features extracted in both time and frequency domain, and parametric and non-parametric classifiers. Preliminary results demonstrate that the classification of activities can be done with remarkable accuracy ( $> 95\%$ ).

**Keywords:** Accelerometer, Android, Smartphone, Activity classification, Quantified-Self

# *Agradecimentos*

A realização desta dissertação marca o fim de uma importante etapa da minha vida. Agradeço a todos aqueles que me apoiaram e contribuíram para que conseguisse fazer este trabalho.

Gostaria de apresentar os meus agradecimentos, acima de tudo, ao Prof. Doutor André Ribeiro Lourenço e ao Prof. Doutor Arnaldo Abrantes por todo o apoio, disponibilidade e óptimas sugestões ao longo deste trabalho. Sem as suas orientações, o sucesso desta investigação não teria sido possível.

Deixo também uma palavra de agradecimento à Ana Priscila Alves do Instituto de Telecomunicações (IT) pela assistência prestada na *framework* e pelos dados fornecidos.

Uma nota de agradecimento a Nuno Mateus, Pedro Poseiro, Sara Ribas, Catarina Vieira, Sofia Carvalho, e à minha irmã Ana Luísa Duarte pela ajuda, paciência demonstrada e conselhos sábios.

Finalmente, agradeço aos meus pais por me terem proporcionado todas as condições para a minha formação académica.



# Índice

Resumo	ii
Abstract	iii
Agradecimentos	iv
Lista de Figuras	vii
Lista de Tabelas	ix
Abreviaturas	xi
Símbolos	xiii
<b>1 Introdução</b>	<b>1</b>
1.1 O Problema . . . . .	2
1.2 Motivação e Contexto . . . . .	2
1.3 Objetivos e Contribuição Original . . . . .	4
1.4 Organização do Documento . . . . .	6
<b>2 Enquadramento e Conceitos</b>	<b>7</b>
2.1 Motivação para uma vida mais ativa . . . . .	9
2.2 <i>Smartphones</i> e Android . . . . .	11
2.2.1 Sistemas Operativos . . . . .	11
2.2.2 Distribuição dos dispositivos Android pela versão, tamanho e densidade dos ecrãs . . . . .	14
2.2.3 Arquitetura Android . . . . .	16
2.3 Sensores e Acelerómetro . . . . .	19
2.3.1 Tipo de Sensores . . . . .	19
2.3.2 Acelerómetro . . . . .	20
2.3.3 Sinal do Acelerómetro . . . . .	22
2.3.4 Frequência de Amostragem em vários modelos de <i>smartphones</i>	23
2.4 Recentes Estudos sobre acelerómetros e deteção de atividades . . .	24
2.5 Produtos e aplicação semelhantes . . . . .	25

---

<b>3</b>	<b>Aplicação Android</b>	<b>31</b>
3.1	Visão geral . . . . .	32
3.1.1	Interface gráfica da aplicação . . . . .	33
3.1.2	Servidor . . . . .	34
3.1.3	Troca de Informação . . . . .	34
3.2	Arquitetura do Sistema . . . . .	35
3.2.1	Blocos Funcionais . . . . .	35
3.2.2	Linguagem no processamento de informação . . . . .	36
3.2.3	Bloco Acelerómetro . . . . .	37
3.2.4	Workflow Manager . . . . .	38
3.2.5	Interface do utilizador . . . . .	39
3.2.6	Comunicação Cliente-Servidor . . . . .	41
<b>4</b>	<b>Classificação automática de atividade</b>	<b>43</b>
4.1	Aquisição do sinal . . . . .	45
4.2	Segmentação do Sinal . . . . .	46
4.3	Extração de características . . . . .	46
4.4	Classificação . . . . .	49
<b>5</b>	<b>Estudo da atividade Andar/Correr</b>	<b>53</b>
5.1	Enquadramento . . . . .	53
5.2	Estudo exploratório . . . . .	55
5.3	Informações sobre a atividade praticada . . . . .	56
<b>6</b>	<b>Análise de resultados</b>	<b>61</b>
6.1	Sinal Adquirido . . . . .	61
6.2	Exploração de Características . . . . .	63
6.3	Resultados da Classificação . . . . .	67
6.4	Informação sobre a atividade . . . . .	72
<b>7</b>	<b>Conclusão e Trabalho Futuro</b>	<b>75</b>
	<b>Bibliografia</b>	<b>77</b>

# Lista de Figuras

2.1	Evolução da ideia proposta . . . . .	8
2.2	Índice de Massa Corporal Médio ( $kg/m^2$ ) da população mundial com mais 20 anos. (Dados de 2008 da OMS) . . . . .	10
2.3	Evolução da venda mundial de <i>smartphones</i> . . . . .	11
2.4	Número relativo de dispositivos que executam uma determinada versão da plataforma <i>Android</i> . . . . .	15
2.5	Arquitetura do Android . . . . .	16
2.6	Eixos do acelerometro . . . . .	22
2.7	Sinal do acelerómetro adquirido através de um <i>smartphone</i> . . . . .	23
2.8	Imagens do <i>layout</i> da aplicação RunKeeper. . . . .	26
2.9	Imagens do <i>layout</i> da aplicação Talos Nike + Running. . . . .	27
2.10	Imagens do <i>layout</i> da aplicação Talos Rowing. . . . .	28
2.11	Fitbit . . . . .	29
2.12	BodyMedia Armband . . . . .	29
2.13	Nike+ FuelBand . . . . .	30
3.1	Arquitetura da framework utilizada. . . . .	32
3.2	Captura de ecrã da aplicação realizada neste projeto . . . . .	41
3.3	Comunicação Cliente/Servidor. . . . .	41
3.4	Figura esquemática da ligação através do protocolo <i>Handshake</i> . . . . .	42
4.1	Arquitetura do sistema proposto . . . . .	44
4.2	Representação esquemática do método de validação cruzada com 5-fold . . . . .	51
4.3	Atividades consideradas neste estudo . . . . .	52
5.1	Ciclo de andamento . . . . .	54
5.2	Algoritmo <i>Peak Detector</i> aplicado a um segmento do sinal adquirido pelo <i>smartphone</i> . . . . .	57
6.1	Módulo do sinal adquirido pelo acelerómetro do <i>smartphone</i> em cada atividade. . . . .	62
6.2	Histograma representando a frequência de amostragem . . . . .	63
6.3	Média e desvio padrão de um conjunto de características estatísticas no domínio do tempo. . . . .	64
6.4	Média e desvio padrão da característica SMA de cada atividade. . . . .	64

---

6.5	Média e desvio padrão dos três elementos de correlação de cada atividade. . . . .	65
6.6	Distribuição das bandas de potência de cada atividade (0-50Hz) . .	66
6.7	Valores médios das bandas de frequência de cada atividade. . . . .	67
6.8	Matriz de confusão do classificador GNB para as características do domínio do tempo. . . . .	71
6.9	Matriz de confusão do classificador Árvores de Decisão para as características do domínio da frequência. . . . .	72
6.10	Algoritmo de detecção de picos aplicado a um segmento de 60 segundos da atividade Andar Outdoor. . . . .	73

# Lista de Tabelas

2.1	Dispositivos com uma determinada configuração de ecrã definido pela combinação entre tamanho e densidade. . . . .	15
2.2	Informação do acelerómetro e da frequência de amostragem adquirida em vários <i>smartphones</i> . . . . .	24
4.1	Especificações do modelo de acelerómetro BMA220 . . . . .	45
5.1	Equivalentes metabólicos genéricos das atividades consideradas neste estudo . . . . .	59
6.1	Resultados da classificação no domínio do tempo. . . . .	69
6.2	Resultados da classificação no domínio da frequência. . . . .	69
6.3	Resultados da classificação no domínio do tempo e da frequência. . . . .	70
6.4	Informação sobre a atividade Andar Outdoor a cinco aquisições de sinal com duração de 60 segundos. . . . .	73



# Abreviaturas

<b>API</b>	<b>A</b> pplication <b>P</b> rogramming <b>I</b> nterface
<b>ASIC</b>	<b>A</b> pplication-specific <b>I</b> ntegrated <b>C</b> ircuit
<b>BSD</b>	<b>B</b> erkeley <b>S</b> oftware <b>D</b> istribution
<b>CSS</b>	<b>C</b> ascading <b>S</b> tyle <b>S</b> heets
<b>DPL</b>	<b>D</b> ata <b>P</b> rocessing <b>L</b> anguage
<b>GNB</b>	<b>G</b> aussian <b>N</b> aive <b>B</b> ayes
<b>GPS</b>	<b>G</b> lobal <b>P</b> ositioning <b>S</b> ystem
<b>GUI</b>	<b>G</b> raphical <b>U</b> ser <b>I</b> nterface
<b>HTML</b>	<b>H</b> yperlink <b>T</b> ext <b>M</b> arkup <b>L</b> anguage
<b>IMC</b>	<b>Í</b> ndice <b>M</b> assa <b>C</b> orporal
<b>JS</b>	<b>J</b> ava <b>S</b> cript
<b>JSON</b>	<b>J</b> ava <b>S</b> cript <b>O</b> bject <b>N</b> otation
<b>KNN</b>	<b>K</b> <b>N</b> earest <b>N</b> eighbor
<b>MEMS</b>	<b>M</b> icro <b>e</b> lectrom <b>e</b> chanical <b>S</b> ystems
<b>METs</b>	<b>M</b> etabolic <b>E</b> quivalents
<b>MPP</b>	<b>M</b> essage <b>P</b> assing <b>P</b> rotocol
<b>MVC</b>	<b>M</b> odel <b>V</b> iew <b>C</b> ontroller
<b>OMS</b>	<b>O</b> rganização <b>M</b> undial da <b>S</b> aúde
<b>SD</b>	<b>S</b> ecure <b>D</b> igital
<b>SDK</b>	<b>S</b> oftware <b>D</b> evelopment <b>K</b> it
<b>SMA</b>	<b>S</b> ignal <b>M</b> agnitude <b>A</b> rea
<b>SO</b>	<b>S</b> istema <b>O</b> perativo
<b>SSL</b>	<b>S</b> ecure <b>S</b> ockets <b>L</b> ayer
<b>SVM</b>	<b>S</b> upport <b>V</b> ector <b>M</b> achine

<b>TCP</b>	<b>T</b> ransmission <b>C</b> ontrol <b>P</b> rotocol
<b>UI</b>	<b>U</b> ser <b>I</b> nterface
<b>VM</b>	<b>V</b> irtual <b>M</b> achine
<b>WFM</b>	<b>W</b> orkflow <b>M</b> anager



# Símbolos

$m$	Metros
$s$	Segundos
$a$	Aceleração
$P$	Potência da densidade espectral
$R$	Matriz de correlação
$SMA$	Área da magnitude do sinal
$S$	Entropia espectral
$ICM$	Índice de massa corporal
$X^l$	<i>Dataset</i> de treino
$y^l$	Legendas do <i>dataset</i> de treino
$X^t$	<i>Dataset</i> de teste
$y^t$	Legendas do <i>dataset</i> de teste
$\sigma$	Desvio Padrão
$\rho$	Elemento de correlação
$ \sigma^i $	Desvio padrão do módulo
$ a $	Média do Módulo



# Capítulo 1

## Introdução

Numa sociedade que cada vez mais percebe os benefícios da prática desportiva para a saúde, o presente estudo tem como principal objetivo contribuir ativamente para o combate ao sedentarismo da população, motivando e monitorizando, através do aproveitamento do sensor acelerómetro, a atividade física.

Desde o início da comercialização dos dispositivos móveis, datada de finais do século XX, que se alcançou um nível tecnológico que não passa despercebido ao comum cidadão. Segundo vários estudos, o desenvolvimento ao nível da computação, dimensão e design, fez com que estes dispositivos se tornassem indispensáveis no dia a dia dos seus utilizadores que, independentemente da faixa etária em que se encontram, se têm vindo a fidelizar cada vez mais <sup>1</sup>.

Os *smartphones* desempenham hoje um papel ubíquo no acesso e processamento de informação. Na sua maioria, possuem uma miríade de sensores integrados (luminosidade, proximidade, aceleração, giroscópios, bússola, GPS) e um potencial de comunicação com a *cloud* que os tornam capazes de gerar informação de elevada precisão e rigor. Este conjunto de factores, tanto de natureza tecnológica como emotiva, está na origem de uma recente tendência de mercado, orientada para o desenvolvimento de plataformas e sistemas operativos, focados nos mais diversos campos de atividade dos seus utilizadores.

---

<sup>1</sup>De acordo com as últimas previsões da ABI Research, o uso de *smartphones* em todo o mundo chegará a 1,4 bilhões este ano [1].

O aproveitamento dos sensores inerentes a estes dispositivos é, de resto, uma das áreas que mais promete ao nível de interação e personalização por parte do utilizador, e que ainda se encontra por explorar ou está a iniciar-se.

## 1.1 O Problema

O sedentarismo é uma preocupação séria que afecta a população sobretudo nos países mais desenvolvidos. As doenças cardiovasculares, as depressões e a obesidade são só algumas das consequências desta prática. Não obstante, a consciência e consequente prevenção para este problema tem crescido em parte devido ao aumento de ferramentas que possibilitam a medição da atividade física.

O controlo e medição da atividade física, num passado próximo, apenas eram possíveis em locais onde a sua prática é uma constante, como o caso dos ginásios ou em clínicas especializadas para o efeito. Neste sentido, não sendo gratuito o acesso a estes locais, o desenvolvimento e o aumento da oferta de dispositivos que permitem fazer esta medição tem sofrido um crescente no último ano. De forma a reduzir os riscos e problemas que podem resultar do sedentarismo, é necessária uma estratégia assente numa comunidade mais consciente e pró-ativa. Essas razões estão na base de um novo conceito, o “Quantified-self”, que consiste na auto-monitorização com o objetivo da auto-consciencialização e aperfeiçoamento pessoal.

No âmbito do nosso trabalho essa estratégia baseia-se na adequação dos *smartphones* para a atividade física, uma vez que estes estão massificados e a sua utilização para a prática desportiva não coloca, a priori, restrições no seu uso.

## 1.2 Motivação e Contexto

Grande parte da população de países desenvolvidos apresenta um estilo de vida com um nível de atividade física abaixo do expectável para a sua faixa etária,

passando a maior parte do dia em ambientes *indoor*. Nesses países, os riscos associados a um estilo de vida sedentário, tais como doenças cardiovasculares, hipertensão, depressão, obesidade, doenças cerebrais e morte prematura [2], tem levado a um aumento do número de indivíduos a praticar desporto, o que exerce influência direta no interesse de aplicações relacionadas com a sua atividade física.

A monitorização e classificação das atividades praticadas apresenta-se como uma forma eficaz de aumentar e motivar a prática desportiva, permitindo ao utilizador quantificar a atividade física realizada e medir o seu progresso. A distinção de atividades é importante uma vez que as atividades têm naturezas completamente diferente e um correspondente consumo energético.

O uso cada vez mais alargado dos *smartphones* tem vindo a aumentar a oferta das aplicações para este efeito. No entanto, na sua maioria, estas aplicações são apenas utilizadas para a monitorização da atividade física através do sensor de posicionamento geográfico (GPS), obrigando o indivíduo, previamente, a escolher a atividade que vai realizar. Não obstante esta obrigatoriedade, este sensor tem também o inconveniente de apenas permitir a monitorização da atividade em ambientes *outdoor* devido à fraca, ou inexistente, recepção de sinal em ambientes *indoor*.

Com este projeto pretendemos, através do uso do acelerómetro, procurar uma forma menos tradicional de monitorizar e classificar as atividades físicas. Devido às suas potencialidades ainda inexploradas, tem havido no último ano um aumento dos estudos relacionados com os acelerómetros assim como as possibilidades e conclusões que se podem extrair do sinal adquirido por estes sensores. O acelerómetro é um sensor presente na grande maioria dos *smartphones* mais recentes e é caracterizado pela sua dimensão e preço reduzidos. Um exemplo comum da sua utilização é a contagem de passos quando andamos ou corremos.

Os acelerómetros são escolhas populares para efetuar a medição do movimento porque têm a capacidade de extrair informação útil para inferir automaticamente a atividade física que um indivíduo está a realizar. Para a classificação de atividades

são usadas ferramentas de aprendizagem automática e inteligência artificial (IA) de forma a permitir a classificação automática de atividades.

No âmbito do nosso trabalho, será desenvolvida uma aplicação que classifica automaticamente a atividade que se está a realizar através do uso de um *smartphone* Android. Foi escolhido este dispositivo por ser um elemento considerado indispensável no dia a dia de grande parte da população, pelas suas abrangentes características e pela taxa de utilização que apresenta.

### 1.3 Objetivos e Contribuição Original

Pretende-se contribuir para a readaptação da oferta de programas considerando as novas tendências e necessidades, tendo como base a análise e apreensão de novas ofertas de serviços.

O principal objetivo deste projeto/dissertação é desenvolver uma aplicação para Android que realize a classificação de atividades através do sinal adquirido pelo acelerómetro do *smartphone* Android. Para realizar essa classificação foi necessário desenvolver algoritmos de classificação automática.

Para conseguir atingir este objetivo foram desenvolvidos módulos para a *framework* Android para que esta realize a aquisição do sinal do acelerómetro e o envie para um servidor localizado na *cloud*, para que desta forma seja possível a análise e classificação da atividade realizada.

No desenvolvimento da aplicação foi utilizada uma nova *framework* de desenvolvimento de aplicações para o Android [3], que, simultaneamente, utiliza as tecnologias Web e permite o rápido desenvolvimento de aplicações.

A classificação é realizada através do método de aprendizagem automática, utilizando uma abordagem supervisionada. O processo da classificação segue a típica cadeia de processamento [4, 5] e está dividido em três partes: aquisição do sinal, extração de características e classificação.

A aquisição do sinal é efetuada com a aplicação desenvolvida, que envia os dados para um servidor na *cloud* para efetuar a classificação. Na segunda parte é realizada a extração de um conjunto de características tanto no domínio do tempo como no domínio da frequência e na terceira parte é realizado um estudo sobre os algoritmos e características mais relevantes para uma classificação mais eficiente.

Este trabalho é inovador nos seguintes aspectos:

- Elaboração de vários módulos de uma *framework* de desenvolvimento de aplicações para Android com o objetivo de adquirir dados do acelerómetro e proceder à classificação de atividade.
- Extensa análise do sinal do acelerómetro adquirido e qual a melhor forma de classifica-lo.
- Realização de vários algoritmos de extração de características.
- Estudo sobre quais as características mais representativas para a eficiente classificação das atividades propostas.
- Estudo sobre quais classificadores que apresentam melhores resultados na classificação.
- Combinação de classificadores para a autenticação da atividade.

Durante a realização deste projeto foi submetido um artigo com o nome de “Activity Classification using a SmartPhone”. O artigo foi aceite para a conferência 2013 IEEE 15th International Conference on e-Health Networking, Applications and Services (Healthcom), que irá ser apresentado em IEEE Healthcom 2013 e será publicado em Proceedings of IEEE Healthcom 2013 e no IEEE eXplore.

Desenvolveu-se ainda um segundo artigo “Activity Classification using a SmartPhone: Multiple users” para a conferência Conference on Electronics Telecommunications and Computers (CETC) que aguarda notificação.

## 1.4 Organização do Documento

O documento está organizado da seguinte maneira:

- No capítulo 1 é descrito de forma pormenorizada as motivações, o contexto, os objetivos e os desafios do trabalho.
- No capítulo 2 é realizado um enquadramento do sistema, apresentado o foco deste projeto assim como o trabalho relacionado nesta área de estudo.
- No capítulo 3 é apresentado o funcionamento da *framework* Android utilizada neste trabalho.
- No capítulo 4 é descrito detalhadamente todo o processo de classificação automática de atividades. Explica como é efetuada a aquisição de sinal, segmentação, extração de características e classificação.
- No capítulo 5 é apresentado um estudo exploratório sobre a estimação de informações sobre as atividades consideradas.
- No capítulo 6 é realizada uma análise e discussão dos resultados obtidos. Este capítulo tem como objetivo perceber o sinal adquirido, quais as melhores características para a classificação assim como quais os melhores classificadores.
- No capítulo 7 são apresentadas as considerações finais, onde é efetuado um balanço sobre todo o trabalho realizado, com especial destaque para os objetivos propostos. O capítulo inclui ainda uma abordagem ao trabalho futuro.



## Capítulo 2

# Enquadramento e Conceitos

Neste capítulo é realizado um enquadramento da ideia desenvolvida e apresentados alguns conceitos fundamentais. A informação é fornecida tendo em consideração a descrição e o objetivo do projeto, de modo que a solução final possa ser apresentada e justificada de forma sequencial com base na revisão literária.

Este projeto foca-se no conceito de *Quantified Self* [6–8]. Este conceito é caracterizado por ser um movimento com o objetivo de incorporar a tecnologia na aquisição de dados em aspectos do dia-a-dia de uma pessoa (por exemplo, comida consumida, qualidade do ar, etc.), estados (por exemplo, humor, excitação, níveis de oxigênio no sangue, etc.) e desempenho (mental e físicos).

O *Quantified Self* apresenta a seguinte metodologia primária para a auto-quantificação: aquisição de dados, seguida da visualização, referenciação cruzada e a descoberta de correlações.

Este conceito não é novo, embora os avanços tecnológicos tenham vindo a ampliar as suas possibilidades. No passado, os métodos de avaliação quantitativa eram penosos e trabalhosos. Nessa altura era necessário inserir os valores manualmente e gravar num registo; retirar medidas manualmente e guardar esses registos num ficheiro; inserir os dados numa folha de cálculo e fazer operações utilizando software pouco intuitivo; e construir gráficos de modo a tentar perceber os números inseridos.

Atualmente, com os avanços tecnológicos a maior parte da aquisição de dados pode ser realizada automaticamente e a manutenção desses registos e análise pode ser realizada com o uso de ferramentas mais intuitivas. Desse modo é possível cada pessoa conhecer-se a si mesma de uma nova forma, permitindo chegar a conclusões que de outra forma eram difíceis.

Os avanços tecnológicos tornaram a aquisição de dados mais acessível e disponível permitindo quantificar por exemplo sinais biométricos, níveis de insulina ou colesterol, batimentos cardíacos, entre muitos outros.

O conceito de *self-tracking*, de registar informação pessoal regularmente, está a proporcionar uma mudança de atitude de cada pessoa perante a sua saúde.

Este projeto exploratório teve como origem a evolução de um sistema que possa vir a motivar e aumentar a atividade física dos seus utilizadores (figura 2.1). Partindo desse pressuposto, e tirando partido das ferramentas que estão atualmente disponíveis no mercado, o projeto evoluiu para o uso dos *smartphones* e de um sensor específico, o acelerómetro. A escolha deste sensor deve-se a este permitir monitorizar e classificar as atividades físicas, temas desenvolvidos pormenorizada-mente nos próximos capítulos.

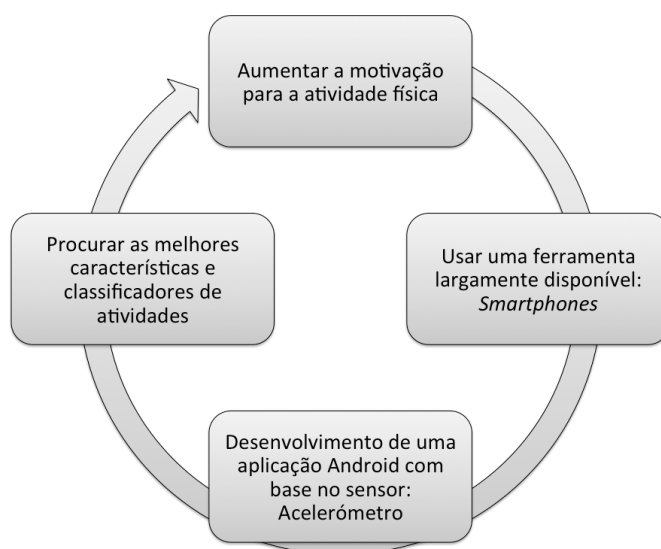


FIGURA 2.1: Evolução da ideia proposta

## 2.1 Motivação para uma vida mais ativa

A inatividade é uma preocupação séria que afecta principalmente os países mais desenvolvidos. Segundo os dados da Organização Mundial da Saúde, aproximadamente 31% dos adultos com mais de 14 anos tiveram atividade física insuficiente, sendo que a percentagem de incidência da inatividade física é de 28% no sexo masculino enquanto que sexo feminino é de 34%. Aproximadamente 3.2 milhões de mortes todos os anos é atribuída ao sedentarismo e aos risco que daí provêm [9].

Os atuais níveis de inatividade física devem-se em parte à realização insuficiente de atividades nas horas de lazer e um aumento de comportamentos sedentários durante as atividades ocupacionais e domésticas. O aumento do uso de modos de transportes considerados passivos contribui também para uma redução do nível de atividade física.

O crescimento dos grandes centros urbanos contribui com vários factores ambientais que podem desencorajar a participação na atividade física, tais como, violência, grande concentração de tráfego rodoviário, baixa qualidade do ar, poluição e a falta de locais que promovam a prática dessas atividades [10].

Para os adultos com idades compreendidas entre os 18 e os 64 anos, a atividade física inclui tempo de lazer (como por exemplo: andar, dançar, nadar, ginásio), transportes (como por exemplo: andar e bicicleta), ocupacional (como por exemplo: trabalho), tarefas domésticas, jogos, desportos ou exercícios planeados no contexto de atividades diárias, família e comunidade.

Como benefícios da atividade física, e comparando os adultos menos ativos com os que levam uma vida mais ativa é de salientar que a atividade física pode estar relacionada com menores taxas de mortalidade, doenças coronárias, hipertensão arterial, acidentes vasculares cerebral, diabetes tipo 2, síndrome metabólico, cancro do cólon e da mama e depressões. Tendem também a ter menos riscos no quadril ou fratura vertebral, apresentam uma melhor aptidão cardiorrespiratória e muscular

e têm maiores probabilidades de atingir a manutenção do peso, ter uma massa corporal e composição saudável.

Outro problema relacionado com a inatividade é o excesso de peso e a obesidade, que são definidas como anormal ou excessiva acumulação de gordura que apresenta risco para a saúde. Como forma de medição da obesidade é utilizada uma medida internacional, chamada de índice de massa corporal (IMC). O IMC é determinado pela divisão da massa do indivíduo pelo quadrado da sua altura, onde a massa está em quilogramas e a altura está em metros. Uma pessoa que tenha um IMC de 30 ou mais é considerada obesa, enquanto que uma pessoa que tenha IMC igual ou maior que 25 é considerada acima do peso [11, 12].

$$IMC = \frac{massa(Kg)}{altura[m] \times altura[m]} \quad (2.1)$$

Na figura 2.2 é apresentado o índice médio de massa corporal da população mundial, segundo os dados de 2008 da Organização Mundial da Saúde [13]. Verifica-se ao observar a figura uma prevalência de excesso de peso e obesidade no continente Americano e Europeu, destacando a América do Norte como a região onde existe a maior taxa de obesidade. O continente Africano e o Sudoeste do continente Asiático são as regiões que apresentam a mais baixa taxa de ICM.

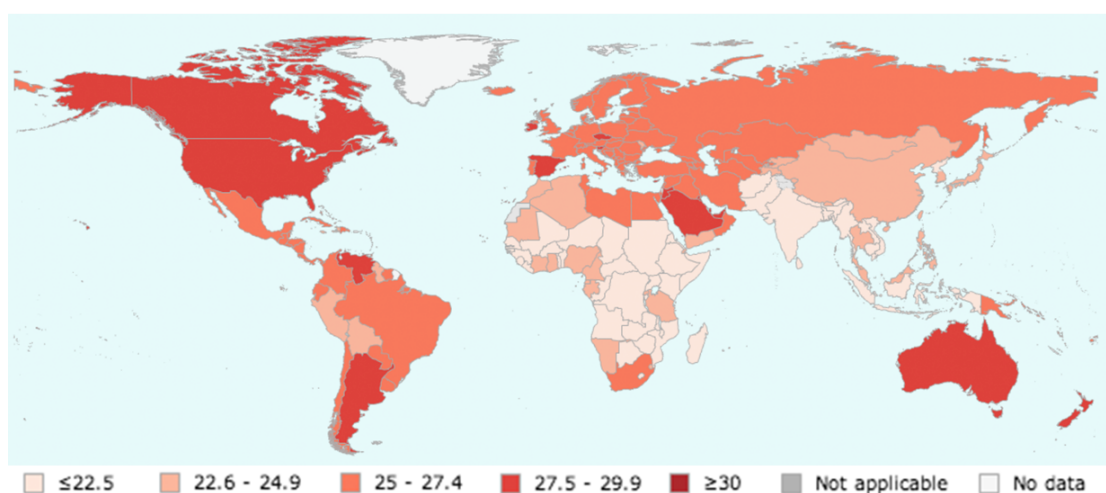


FIGURA 2.2: Índice de Massa Corporal Médio ( $kg/m^2$ ) da população mundial com mais 20 anos. (Dados de 2008 da OMS)

## 2.2 *Smartphones e Android*

O mercado dos *smartphones* tem sido conformado com grandes alterações nos últimos anos. Segundo os dados da Gartner [14] é apresentado na figura 2.3 a evolução das vendas de *smartphones* no mercado mundial por sistema operativo. Observando a figura tenta-se perceber a evolução dos sistemas operativos assim como perceber qual a tendência.

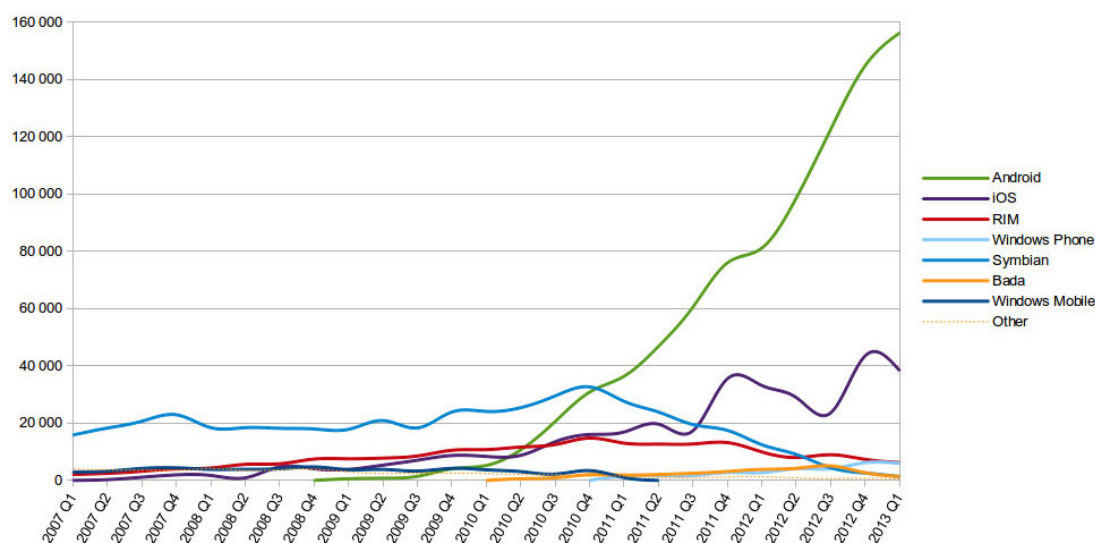


FIGURA 2.3: Evolução da venda mundial de *smartphones* (Fonte: Gartner)

Conclui-se a partir da figura 2.3 que a venda de *smartphones* com o sistema operativo Android apresenta um crescimento exponencial destacando-se dos seus concorrentes a partir de 2011.

### 2.2.1 *Sistemas Operativos*

Dentro dos sistemas operativos mais comuns estão o Android da Google, Bada da Samsung Electronics, BlackBerry OS da RIM, iOS da Apple Inc., Windows Phone e Windows Mobile da Microsoft e Symbian OS da Nokia e Accenture.

#### **Android**

Android é uma plataforma gratuita e *open-source*, derivada do Linux, fundada em outubro de 2003 por Andy Rubin e adquirida em 2005 pela Google.

Em 2007 é anunciada a Open Handset Alliance, consórcio de empresas com o objetivo de desenvolver padrões abertos para dispositivos móveis, onde o SO é oficialmente tornado num software *open-source*.

Em 2008, o Android SDK 1.0 é lançado. O equipamento HTC Dream, também conhecido como T-Mobile G1, é vendido pela operadora sem fio T-Mobile nos EUA e é o primeiro *smartphone* comercializado a correr o sistema operativo Android.

Em 2009 assiste-se a um aumento dos dispositivos baseados em Android e o lançamento de novas versões do sistema operativos:

- Android 1.0 - Setembro de 2008
- Android 1.1 - Fevereiro de 2009
- Android 1.5 Cupcake - Abril de 2009
- Android 1.6 Donut - Setembro de 2009
- Android 2.0/2.1 Eclair - Outubro de 2009 / Janeiro de 2010
- Android 2.2.x Froyo - Maio de 2010
- Android 2.3.x Gingerbread - Dezembro de 2010
- Android 3.x Honeycomb - Fevereiro de 2011
- Android 4.0.x Ice Cream Sandwich - Outubro de 2011
- Android 4.1/4.2/4.3 Jelly Bean - Junho de 2012
- Android 4.4 KitKat - Outubro de 2013
- Android 5.0 Key Lime Pie - Ainda não foi lançada.

## **Bada**

O sistema operativo Bada foi lançado pela Samsung em 2009. O primeiro *smartphone* lançado pela marca, chamado de Samsung Wave, foi lançado mais tarde em 2010. A Samsung usa o sistema operativo Bada ao mesmo tempo que o sistema operativo Android e Windows Phone, mas em 2012 os smartphones que usam o Bada

foram discretamente alterados em favor da utilização do sistema operativo Android. Os dispositivos comercializados com o SO *Bada* são comercializados com a marca *Wave* e todos os dispositivos que são comercializados com o SO *Android* são comercializados com a marca *Galaxy* [15, 16].

### **BlackBerry OS**

O sistema operativo da BlackBerry é um sistema proprietário desenvolvido pelo *Research In Motion* para a linha de *smartphones* da BlackBerry [17]. Este sistema permite suporte nativo para o ambiente de comunicação corporativa, que permite um acesso wireless e sincronização do e-mail, calendário, tarefas, notas e contactos. Atualmente os dispositivos BlackBerry incluem os sistemas operativos v5.0, v6 e v7. O BlackBerry 10 SO é um sistema operativo baseado em QNX, similar ao encontrado nos tablets RIM's Playbook, sistema que irá substituir o SO BlackBerry em 2013.

### **iOS**

O iOS é um sistema operativo desenvolvido e distribuído pela Apple Inc [18]. Inicialmente chamado de iPhone OS foi lançado em 2007 para o iPhone e iPod Touch. Depois de alguns anos, foi estendido para suportar outros dispositivos da Apple como é o caso do iPad e da Apple TV e renomeado para iOS. Ao contrário dos outros sistemas operativos como o Android da Google's ou o Windows Phone da Microsoft, a Apple não distribuí a licença do iOS para os outros fabricantes sendo o único fabricante de dispositivos com o sistema operativo iOS.

### **Windows Phone e Windows Mobile**

Windows Mobile é o antecessor do Windows Phone. Em 2010 a Microsoft anunciou que iria descontinuar o Windows Mobile, e que este SO iria ser incompatível com o novo SO disponibilizado [18]. O Windows Mobile tinha duas edições específicas, o Windows Mobile Standard e Windows Mobile Professional, sendo que ambas as edições incompatíveis uma com a outra. O Windows Mobile é um produto desenvolvido pela Microsoft mas a empresa não produziu nenhum dispositivo, apenas vendia as licenças para os fabricantes de hardware. Em Outubro de 2010 foi lançado o Windows Phone 7 seguido do Windows Phone 7.5 Mango, Windows

Phone 7.5 Tango, Windows Phone 7.8 e Windows Phone 8. Este último foi lançado em Outubro de 2012, sendo introduzidas características muito aguardadas, como o multi-core e suporte de múltiplas telas, suporte a cartões de memória, o modo de armazenamento em massa USB, suporte de produção de vídeo com os formatos DivX / Xvid, suporte a NFC, transferências de arquivos Bluetooth, entre outras funções.

### **Symbian OS**

O SO Symbian foi desenvolvido pela Symbian Corporation e Nokia, lançado em 1997 [18]. Este sistema foi projetado para dispositivos móveis, foi líder na plataforma de *smartphones* de 2003 até 2010. Depois dessa data, o sistema operativo da Google Android assumiu a liderança.

## **2.2.2 Distribuição dos dispositivos Android pela versão, tamanho e densidade dos ecrãs**

As várias versões do Android são distribuídas em números bastante diferentes, ver figura 2.4.

Os dados recolhidos no website do Android Developers [19], durante um período de 14 dias que acabou em 1 de Agosto de 2013, demonstra que as versões que têm o maior número de dispositivos são a Jelly Bean com 40,5%, Gingerbread com 33.0% e Ice Cream Sandwich com 22.5%. Todas as versões com uma distribuição inferior a 0,1% não são consideradas na figura 2.4.



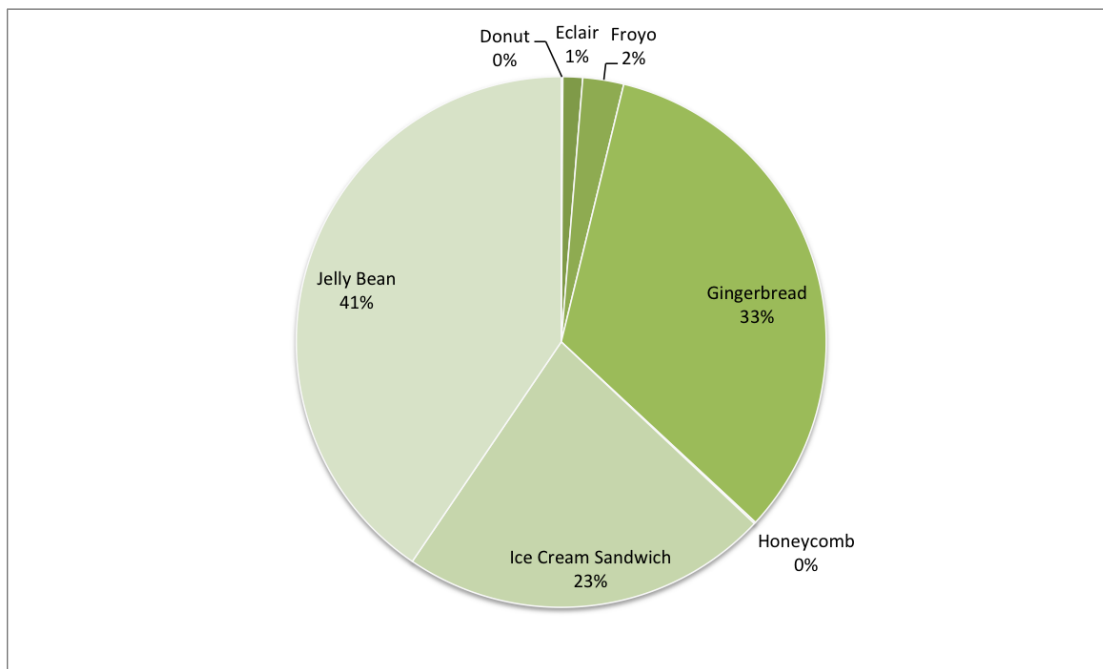


FIGURA 2.4: Número relativo de dispositivos que executam uma determinada versão da plataforma *Android*

Os vários *smartphones* têm várias configurações de ecrã, definidos pela combinação do tamanho do ecrã com a densidade. O SO Android para facilitar o design da interface para diferentes configurações de ecrãs são divididos em vários tipos de configurações, são essas em tamanho Small, Normal, Large, XLarge e de densidade: Ldpi, mdpi, tvdpi, hdpi, xhdpi e xxhdpi [19].

	lpdi	mdpi	tvdpi	hdpi	xhdpi	xxhdpi	Total
Pequeno	9,7 %	0,0 %	0,0 %	0,0 %	0,1 %	0,0 %	9,8 %
Normal	0,1 %	15,9 %	0,0 %	34,5 %	23,9 %	5,7 %	80,1 %
Large	0,5 %	3,2 %	1,1 %	0,4 %	0,5 %	0,0 %	5,7 %
XLarge	0,0 %	4,1 %	0,0 %	0,2 %	0,1 %	0,0 %	4,4 %
Total	10,3 %	23,2 %	1,1 %	35,2 %	24,5 %	5,7 %	

TABELA 2.1: Dispositivos com uma determinada configuração de ecrã definido pela combinação entre tamanho e densidade.

### 2.2.3 Arquitetura Android

O Android é um conjunto de software para dispositivos móveis que inclui um sistema operativo, *middleware* e aplicações chave. É disponível um SDK que fornece ferramentas e APIs necessárias para começar a desenvolver aplicações na plataforma Android usando a linguagem de programação Java.

Como se pode observar na figura 2.5 a arquitetura do Android pode ser subdividida em cinco camadas, são essas, o Kernel e as ferramentas de baixo nível, as bibliotecas nativas, o runtime do Android, a framework, e no topo de todas as camadas, as aplicações [20].



FIGURA 2.5: Arquitetura do sistema operativo Android (Fonte: Android Developers)

Camadas da Arquitetura do Android:

- **Kernel:** A layer base do sistema Android é o Kernel, em cima do qual todo o sistema Android é construído. O sistema Android tem como base a versão

do Linux 2.6, sendo este o sistema que interage com o hardware e contém todos os *drivers* que permitem o controlo e comunicação com o hardware, essenciais para o funcionamento do sistema. O Kernel atua como camada de abstração entre o hardware e a camada de software, e é utilizado nos sistema Android como funções base como é o caso da segurança, gestão de memória, gestão de processos, gestão de tarefas e módulos de *drivers*.

- **Bibliotecas:** As bibliotecas, escritas nas linguagens C e C++, são específicas para determinado hardware, e permite o dispositivo controlar diferentes tipos de dados. Essas capacidades de controlo de dados são apresentadas aos *Developers* através da framework do Android. Em baixo são apresentadas as bibliotecas base para o funcionamento do sistema:
  - SQLite: SQLite é um motor de base de dados usado no Android com o objetivo de guardar dados.
  - Surface Manager: Faz a gestão do acesso ao subsistema de exibição bem como as múltiplas camadas de aplicações 2D e 3D.
  - WebKit: É o motor de pesquisa usado para mostrar conteúdo HTML.
  - SGL: O subjacente motor gráfico 2D.
  - Media framework: Fornece diferentes *codecs* que permite a gravação e reprodução de diferentes formatos multimédia. Permite o suporte de formatos comuns de áudio, vídeo e imagens (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF).
  - OpenGL: Usado para render conteúdo gráficos 2D e 3D para o ecrã.
  - FreeType: Permite render imagens *bitmap* e vectores de fontes.
  - SSL: Biblioteca que permite a configurações de segurança.
  - Libc: O Android usa uma biblioteca escrita em C, chamada de Bionic, que tem uma pequena API concebida para ser embebida em dispositivos baseados em Linux.
- **Android Runtime:** O Android Runtime consiste na máquina virtual Dalvik e nas livrarias base da linguagem de programação Java. A máquina virtual Dalvik é um interpretador que permite ao dispositivo correr múltiplas

máquinas virtuais de modo eficiente, sendo que cada aplicação corre o seu próprio processo e cria uma nova instância da máquina virtual Dalvik. A VM Dalvik executa os arquivos executáveis Dalvik (.dex) formato que é otimizado para o baixo consumo de memória. Esta VM depende do kernel do Linux para a funcionalidade subjacente como a gestão de baixo nível de threads e de memória.

- **Framework:** A *layer Framework* permite um nível de abstração promovendo o desenvolvimento de aplicações novas e inovadoras. Fornece o acesso a blocos, aos quais as aplicações acedem diretamente, dando aos utilizadores a possibilidade de tirar partido das vantagens do hardware do dispositivo, o acesso a informação de localização, assim como a gestão das atividades, da partilha de dados entre aplicações, entre outras. A *framework* foi construída de modo a facilitar o desenvolvimento de aplicações permitindo a reutilização de componentes e a partilha de capacidades entre aplicações.
- **Aplicações:** Encontra-se no topo das camadas da arquitetura Android e é onde as aplicações do *smartphone* vão pertencer. As aplicações vão correr nas suas *sandboxed* Dalvik VM e podem consistir em múltiplas componentes como é o caso das atividades, serviços, recepção de *broadcast* e de fornecedores de conteúdos. Os componentes podem interagir com outras componentes da mesma ou de diferentes aplicações via *intents*. O sistema Android são comercializados com um conjunto de aplicações base como é o caso de programa para recepção de email, sms, calendário, mapas, *browser*, entre outros. Todas as aplicações são escritas usando a linguagem de programação Java.

## 2.3 Sensores e Acelerómetro

### 2.3.1 Tipo de Sensores

Os *smartphones* estão cada vez capacitados com os mais variados tipos de sensores que, de certa forma, promovem uma experiência mais rica para os seus utilizadores. Esses sensores são capazes de medir o movimento, posicionamento e outras condições ambientais, capazes de produzir dados com elevada precisão e exatidão. Esses sensores em termos físicos são componentes de hardware conectados aos dispositivos de forma a fornecer dados do mundo real para as aplicações [21].

A plataforma Android é constituída por três grandes categorias de sensores, são essas:

- Sensores de movimento: Estes sensores medem as forças de aceleração e de rotação (linear e angular) ao longo dos três eixos. Esta categoria inclui acelerómetros, sensor de gravidade, giroscópio, e sensor de rotação vetorial (a partir de Android 2.3).
- Sensores ambientais: Este sensores medem vários parâmetros ambientais, tais como a temperatura ambiente e a pressão, iluminação e humidade. Esta categoria inclui barómetros, fotómetros e termómetros.
- Sensores de posicionamento: Estes sensores medem a posição física do aparelho. Esta categoria inclui sensores de geoposição (GPS), proximidade e de campos magnéticos.

A maior parte dos *smartphones* têm pelo menos três sensores base, são esses, o acelerómetro de três eixos para medir a aceleração, o magnetómetro de três eixos para medir o campo magnético e um sensor de giroscópio para medir ou obter a orientação.

### 2.3.2 Acelerómetro

O acelerómetro é um componente electrónico integrado que mede as forças de aceleração exercidas num determinado objecto. Essas forças podem ser de dois tipos: estáticas ou dinâmicas, sendo a primeira a força da aceleração gravítica (constante), e a segunda causada pelo movimento ou vibração provocado no acelerómetro. Ao medir a aceleração estática é possível descobrir o ângulo de inclinação do dispositivo em relação à terra, e ao medir a aceleração dinâmica é possível analisar o tipo de movimento exercido num determinado objecto [22, 23].

Existem vários tipos de acelerómetros cada um com as suas próprias características, vantagens e desvantagens. Os acelerómetros são divididos em dois grandes grupos: os mecânicos e ou electromecânicos. No primeiro grupo os acelerómetros mais comuns são os capacitivos, os piezoeléctricos e os piezoresistivos enquanto que no segundo grupo, pode-se destacar os MEMS, sensores atualmente em grande expansão. [23, 24].

Os MEMS, que está na base deste trabalho de investigação, explora as propriedades mecânicas do silicone para criar estruturas móveis que revelam as acelerações e vibrações em direcções distintas e apresentam como grande vantagem serem de custo reduzido, de pequena dimensão, sensíveis e permitirem dar novas funcionalidades aos produtos que os detêm.

O sensor MEMS de um acelerómetro linear tem como base uma estrutura de silicone, com interdigitações em forma de pente, composta por dedos fixos e móveis. A aceleração é obtida da medição dos descolamentos de elementos móveis que estão associados aos eixos, podendo ser de um, dois ou três eixos.

O sinal do acelerómetro com três eixos pode ser descrito da seguinte maneira:

$$a_m = \frac{1}{m}(F - F_g), \quad (2.2)$$

sendo que  $a_m$  é a medida da aceleração,  $m$  é a massa do corpo,  $F$  é a soma das forças exercidas no corpo e  $F_g$  é a força exercida pela gravidade ( $9,8m/s^2$ ), também detectada pelo sensor.

A força da aceleração gravítica está presente no sinal adquirido pelo acelerómetro porque essa força exerce pressão na massa contra o material, mudando a sua capacidade eléctrica. Por essa razão, se o acelerómetro estiver imóvel, a gravidade produz uma tensão para baixo, que aparenta no sinal adquirido ser equivalente a uma aceleração para cima. No caso de o acelerómetro estar em queda livre a aceleração registada será nula, contrariando o facto de este estar efetivamente com uma aceleração de  $9,8m/s^2$ .

Devido ao seu custo, estes sensores apenas estavam disponíveis para indústrias como a aeroespacial ou militar que normalmente estão no vanguarda da desenvolvimento. Mas com a evolução dos sistemas MEMS de baixo custo e tamanho, tornou-se vantajoso a sua utilização noutras indústrias como por exemplo a automóvel, geográfica entre outras.

O acelerómetro é um tipo de sensor atualmente comercializado na grande maioria dos novos dispositivos móveis. O uso mais comum deste sensor nos *smartphones* é a rotação do ecrã, deteção de choque, controlo de jogos com inclinação do ecrã e a execução de operações base como silenciar uma chamada, desligar o despertador, parar ou prosseguir uma música através de simples movimentos, ou alternando o posicionamento do próprio dispositivo como por exemplo posicioná-lo com o ecrã virado para baixo.

Nos *smartphones* o posicionamento padrão do acelerómetro é no centro de massa do dispositivo e utiliza habitualmente um sistema de coordenadas de 3-eixos, como demonstrado na figura 2.6. No caso de ser mantido na orientação padrão, o eixo dos x é horizontal e aponta para a direita, o eixo dos y é vertical e aponta para cima e o eixo dos z é perpendicular ao ecrã, sendo positivo quando aponta para a frente do ecrã.

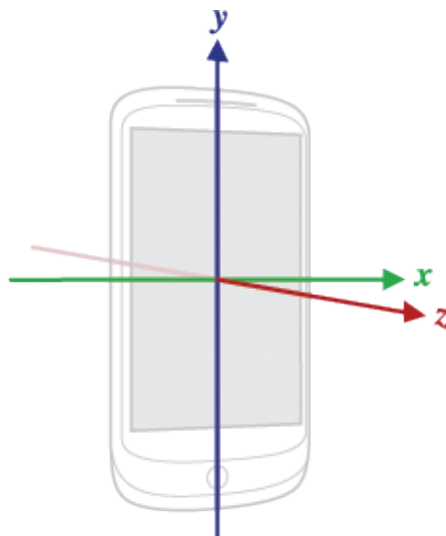


FIGURA 2.6: Demonstração da disposição dos eixos num dispositivo móvel

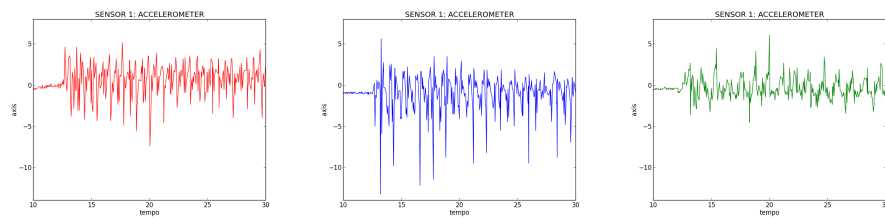
Neste estudo, o acelerómetro mais utilizado foi o modelo BMA220 da BOSCH. É caracterizado por ser um sensor ultra pequeno de três eixos perpendiculares que mede a aceleração *low-g* através de um interface digital, tendo como alvo dispositivos de baixa potência. Funciona através de um circuito de avaliação (ASIC) que converte a saída de uma estrutura de sensor de aceleração microeletromecânico (MEMS) que funciona segundo o princípio de capacitância diferencial.

### 2.3.3 Sinal do Acelerómetro

O sinal do acelerómetro adquirido pelo sistema operativo Android é um array constituído por três valores, cada valor correspondendo a um eixo ( $x$ ,  $y$ ,  $z$ ), estando todos em  $m/s^2$ .

De forma a obter uma visualização do sinal do acelerómetro, adquirido de um indivíduo em andamento com o dispositivo na cintura, o sinal é decomposto nos três eixos ( $x$ ,  $y$  e  $z$ ), correspondendo a componente  $x$  à aceleração frontal, a componente  $y$  a aceleração vertical e a componente  $z$  à aceleração lateral, figura 2.7.





(a) Acelerómetro segundo o eixo x. (b) Acelerómetro segundo o eixo y. (c) Acelerómetro segundo o eixo z.

FIGURA 2.7: Sinal do acelerómetro adquirido através de um *smartphone*

### 2.3.4 Frequência de Amostragem em vários modelos de *smartphones*

De modo a perceber a influência da diferença de hardware e software entre vários modelos e marcas de *smartphones* com o SO Android foi realizado um levantamento de informações de alguns dispositivos. Para esta aquisição foi tido em conta a obtenção das amostras o mais rápido possível, através da opção `SENSOR_DELAY_FASTEST` [25].

Os dados obtidos na tabela 2.2 foram capturados utilizando uma aplicação disponível no Google Play com o nome de *Accelerometer Frequency* e tem como finalidade mostrar informações sobre o acelerómetro assim como a frequência de amostragem obtida [26].

Será importante salientar, que ao observar a figura 2.2, o dispositivo que apresenta a frequência de amostragem média mais baixa é o HTC Wildfire com 44Hz, enquanto que a mais alta é apresentada pelo dispositivo Samsung Galaxy Gio com um valor médio de 91Hz. O desvio padrão mais elevado é apresentado pelo *smartphone* Sony Ericsson com um valor médio de 8,6Hz e a mais baixa é apresentada pelo dispositivo Samsung Galaxy Ace com um valor médio de 2,3Hz.

<b>Modelos</b>	<b>SO</b>	<b>Modelo</b>	<b>Power (mA)</b>	<b>Freq. Mínima (Hz)</b>	<b>Freq. Máxima (Hz)</b>	<b>Freq. Média (Hz)</b>	<b>Desvio Padrão (Hz)</b>
HTC Wildfire	2.2.1	BMA150	3.000	22.07	52.84	44.2	4.1
Samsung Gal. Gio	2.3.3	BMA220	0,005	77.20	101.56	91.5	4.4
Sony Ericsson	2.3.4	BMA250	0,003	62.49	101.54	80.6	8.6
HTC Desire HD	2.3.5	BMA220	0,200	68.15	28.53	54.6	6.6
Samsung Gal. Ace	2.3.6	BMA222	0,100	44.09	65.28	61.1	2.3

TABELA 2.2: Informação do acelerómetro e da frequência de amostragem adquirida em vários *smartphones*

## 2.4 Recentes Estudos sobre acelerómetros e deteção de atividades

Na literatura a maioria dos estudos é baseada no reconhecimento de atividades usando vários acelerómetros, localizados em diferentes partes do corpo. Contudo, este estudo utiliza o acelerómetro de um *smartphone* no bolso frontal do indivíduo, de maneira a tentar ser o mais natural e real possível. Nesta secção é apresentado um resumo dos trabalhos mais significativos na área de deteção e classificação de atividades que têm por base o acelerómetro.

Para o estudo do sinal do acelerómetro destaca-se os trabalhos de Tae-Seong Kim [27, 28], Neil Zhao [29], Tom Mikael Ahola [30].

Na análise dos padrões do sinal e dos classificadores foram considerados os livros: *The Elements of Statistical Learning* [4] e *Pattern Classification* [5].

No estudo das características do domínio do tempo são destacados os artigos [31, 32] e no domínio da frequência [33, 34].

Para o estudo do padrões do comportamento humano foi tido como referência o livro de Roger Bartlett's, onde são abordados pormenorizadamente os comportamentos humanos enquanto se pratica várias atividades desportivas [35].

Para o cálculo do equivalente metabólico associado a cada atividade, destaca-se o artigo "Accelerometer Output and MET Values of Common Physical Activities" [36] e o artigo "Estimating energy expenditure accelerometers" [37].

## 2.5 Produtos e aplicação semelhantes

Como este projeto se foca no estudo da atividade física através do uso do *smartphone* foi realizado um levantamento das aplicações atualmente disponíveis no Android assim como alguns dispositivos que se baseiam no acelerómetro e estão relacionadas com este tema.

### **Run Keeper**

Esta aplicação com suporte para Android e IOS tem como objetivo a gravação, medição e melhoramento do exercício físico [38]. Pode ser utilizado para atividades como correr, caminhar, ciclismo e utiliza o GPS do *smartphone* para visualização do percurso realizado. Na medição do exercício da atividade é possível ver detalhes à cerca do número de passadas, distância, tempo e calorias queimadas, assim como obter estatísticas e progressos através do auricular. Permite ainda medir o batimento cardíaco (usando um acessório), tirar fotografias ao longo do caminho e partilhá-las. Esta aplicação mede a performance ao longo do tempo permitindo visualizar um histograma detalhado das atividades (corrida, caminhada, ciclismo, entre outras) para ver os progressos e receber notificações quando é atingido novo recorde pessoal e metas. Outra opção é seguir um plano detalhado para ajudar a atingir objetivos como por exemplo perda de peso através da corrida e visualizar o percurso realizado no decorrer da atividade. De forma a partilhar com os amigos é possível publicar as suas estatísticas (mapa, distancia, calorias e tempos) e realizações e planos de formação em redes sociais tais como Facebook, Twitter e os

“amigos” no RunKeeper. A aplicação permite ainda integrar as informações das suas atividades com outras aplicações, tais como, Fitbit, Withings, Zeo, Garmin entre outras, para obter uma compreensão mais pormenorizada da saúde no geral.

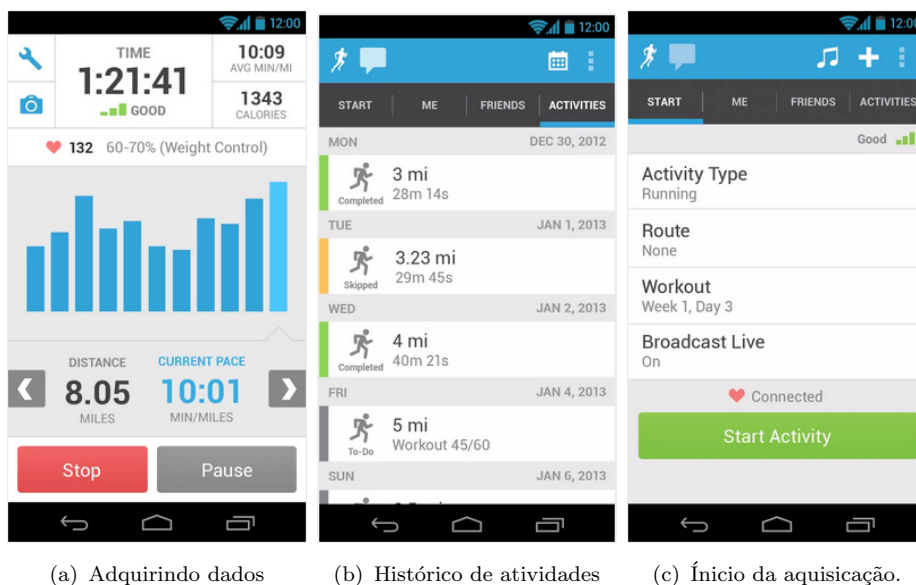


FIGURA 2.8: Imagens do *layout* da aplicação RunKeeper.

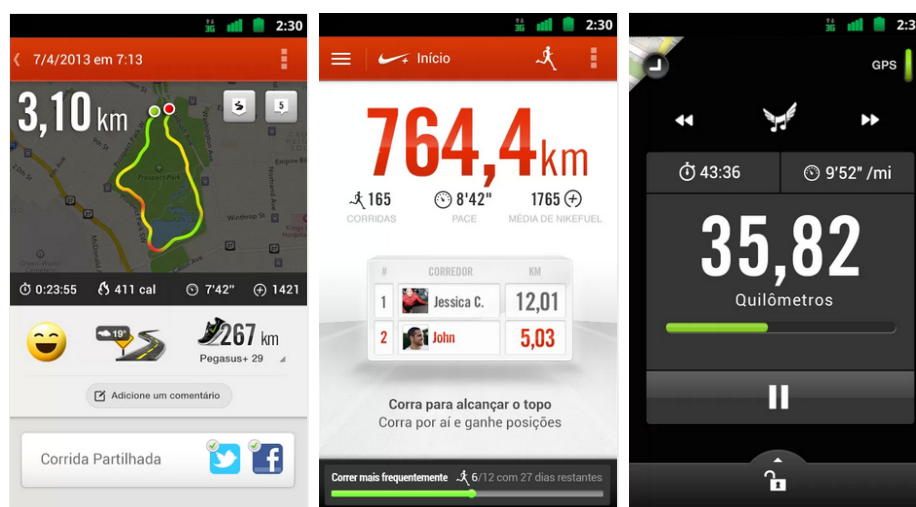
## Runstar

O programa runstar oferece feedback áudio; atualização automática do trabalho realizado e quatro modos de corrida (treadmill, distance, time and free) [39]. A aplicação apresenta incentivos durante a atividade com o objetivo de motivar o utilizador e medir o progresso. Permite ainda a utilização de objetivos (tempo e distância) e a partilha dos resultados com os amigos.

## Nike+ Running

Este programa grava num mapa o percurso percorrido, monitoriza o progresso e envia informações ao utilizador para este ter a motivação necessária para cumprir os objetivos propostos [40]. Com esta aplicação é possível gravar, através do uso do acelerómetro e do GPS a distância percorrida, a passada e o tempo, permitindo ainda através de um feedback áudio informação sobre o exercício realizado de forma a que o utilizador continue focado na corrida. Permite publicar na rede

social Facebook informações relativas ao início da corrida, marcar amigos com quem costuma correr e partilhar o mapa da rota de corrida realizado. Permite ainda obter um resumo com os percursos percorridos e qual a performance, o terreno percorrido assim como as condições climáticas, informações da cadência em cada milha e a comparação com performances praticadas em corridas anteriores. Com a aplicação é possível visualizar todas as corridas de modo a ver o histórico dos progressos num gráfico de barras.



(a) Percurso realizado

(b) Visão global

(c) Adquirindo dados

FIGURA 2.9: Imagens do *layout* da aplicação Talos Nike + Running.

## Running Tracker

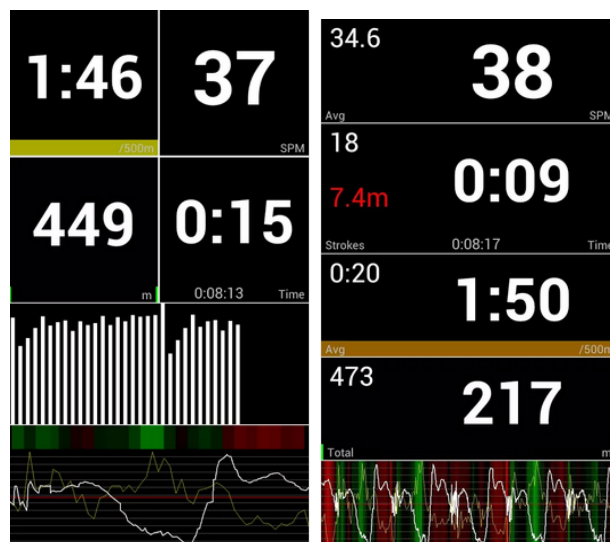
A aplicação Running Tracker é uma aplicação que permite através do sensor do GPS o cálculo da distância percorrida, opções de começar/pausar/recomeçar a corrida, modo *indoor* sem a utilização do GPS, resumos anuais/mensais/semanais/diários, guarda numa base de dados on-line e faz o cálculo e controlo das calorias gastas [41]. Permite o backup e restauro dos dados a partir da Dropbox, Google Drive e cartão sd, exportação dos relatórios para formatos .csv e html, o suporte de várias métricas e ainda filtros de seleção temporal.

## GPS Walk and Run Tracker

Esta aplicação é iniciada no princípio do exercício e informará o utilizador do tempo da atividade, distância percorrida, velocidade média e velocidade máxima [42].

### Talos Rowing

É uma aplicação desenvolvida para a prática de remo [43]. Esta aplicação tem as seguintes características: quantidades de remadas, velocidade, distância, tempo decorrido, tempo parcial e a taxa média das remadas.



(a) Adquirindo dados 1

(b) Adquirindo dados 2

FIGURA 2.10: Imagens do *layout* da aplicação Talos Rowing.

### MapMyRun GPS Running

O MapMyRun+ determina a rota, a distância, a velocidade, o passo as calorias em tempo real usando o GPS [44]. Este programa é indicado para atividades: *fitness*, *jogging*, perda de peso, contagem de calorias queimadas, entre outras.

### Fitbit: ZIP

O ZIO da Fitbit é um dispositivo que tem um acelerómetro de 3 eixos com o objetivo de medir o padrão de movimento de cada utilizador. Permite determinar as

calorias queimadas, distância percorrida e número de passos que foram dados [45]. Embora seja um dispositivo de uso externo permite que seja sincronizado com os *smartphones* para ter a informação atualizada e em tempo real.



FIGURA 2.11: Fitbit

### BodyMedia FIT

O BodyMedia FIT é um dispositivo externo que utiliza 4 sensores, sendo esses, um acelerómetro de 3 eixos, um sensor de temperatura, sensor de fluxo de calor e um sensor que mede a resposta galvânica da pele [46]. Este dispositivo realiza uma monitorização diária das atividades, de forma a promover a perda de peso e a gestão das calorias gastas.



Wireless LINK Armband

FIGURA 2.12: BodyMedia Armband

### Nike + FuelBand

O Nike + FuelBand é um medidor de atividade usado no pulso. Esta pulseira, faz parte do movimento “Quantified-Self”, permitindo aos utilizadores medir a sua

atividade física, os passos realizados e a quantidade de calorias queimadas, através de um acelerómetro. O dispositivo conecta-se via *wireless* à aplicação e possibilita ver os progressos no *smartphone*. Permite desbloquear *achievements* e partilhar informação com os contactos do Facebook e Twitter, de forma a ser encorajado e motivado. Permite ainda comparar e competir com amigos do Facebook.



FIGURA 2.13: Nike+ FuelBand



## Capítulo 3

# Aplicação Android

A aplicação criada neste projeto baseia-se na *framework* MobileBIT [3], criada no Instituto de Telecomunicações, e vai ser apresentada neste capítulo.

A aplicação foi desenvolvida para o sistema operativo Android que é uma versão modificada do SO Linux. Este sistema apresenta as vantagens de estar presente em milhões de aparelhos móveis em todo o mundo, apresentar um rápido crescimento nos últimos anos (figura 2.3) e oferecer a possibilidade de desenvolvimento independentemente da versão, permitindo aos *developers* apenas precisar de desenvolver para Android e a aplicação deverá ser compatível com na maioria dos dispositivos que utiliza esse sistema, independentemente do fabricante [47].

De forma esquemática é apresentado na figura 3.1 a *framework* utilizada para o desenvolvimento da aplicação que efetua a captação e envio do sinal adquirido do acelerómetro para o servidor, onde algoritmos de classificação automática processam esses sinais.

No desenvolvimento da aplicação, o foco principal foi o bloco acelerómetro, responsável pela aquisição, envio e gravação do sinal do sensor.

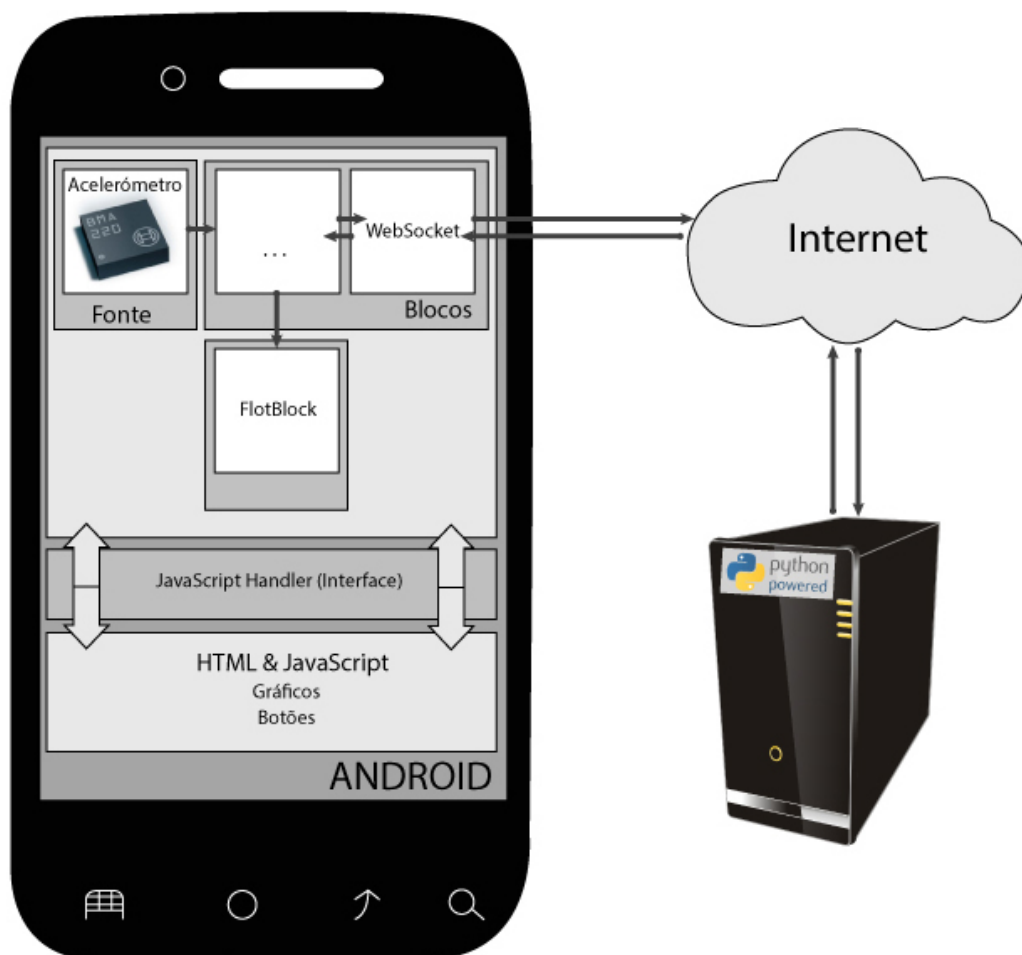


FIGURA 3.1: Arquitetura da *framework* da aplicação utilizada, onde o sinal adquirido pelo acelerômetro do *smartphone* é apresentado na interface gráfica e enviado para um servidor remoto de modo a ter processamento eficiente na classificação de atividade; Os resultados da fase de processamento são retornados de volta para o utilizador.

### 3.1 Visão geral

Nesta secção é efetuada uma breve contextualização dos conceitos principais e das tecnologias usadas na *framework* MobileBit.

### 3.1.1 Interface gráfica da aplicação

O sistema operativo Android disponibiliza uma componente chamada de Webview, que permite mostrar as páginas Web como parte do *layout* das aplicações nativas. Esta classe permite correr um WebBrowser ou apenas mostrar conteúdo online dentro da atividade. Usa ainda um motor de renderização WebKit de forma a exibir páginas Web e incluir métodos para navegação. A componente Webview do Android possibilita a utilização de código JavaScript possibilitando o design e a unificação das interfaces baseada em tecnologias Web.

A interface gráfica é um importante aspecto da aplicação visto ser através dela que o utilizador tira proveito das suas potencialidades. Esta deverá ser intuitiva e promover uma boa experiência de utilização.

Com a evolução da utilização dos Browser Web, hoje suportados pela maioria dos dispositivos móveis, tais como *smartphones* e *tablets*, a solução aqui apresentada tira partido dessa capacidade de modo a permitir uma experiência mais rica para os utilizadores, facilidade no design e compatibilidade multiplataforma.

A tecnologia utilizada foi o HTML, omnipresente na Web, fornecendo meios para criar documentos estruturados pela denotação semântica estrutural do texto, tais como, títulos, parágrafos, listas, links, citações entre outras [48]. Associado ao HTML está o CSS (Cascading Style Sheets), que é uma folha de estilos para estilizar as páginas Web correspondentes. Estes estilos definem como mostrar os elementos HTML e são normalmente gravados em ficheiro externos com a extensão “.css”, permitindo alterar a aparência do site apenas editando esse ficheiro [49].

Neste projeto foi ainda associado ao HTML o JavaScript que ajuda a criar Websites mais dinâmicos e atraentes para o utilizador.

Os últimos standards – HTML5 [50], CSS3 [51] - e bibliotecas – tais como jQuery para o JS [52] – oferecem meios para produzir simples e intuitivas páginas proporcionando as melhores experiências para os seus utilizadores.

### 3.1.2 Servidor

Esta aplicação, como demonstrado na figura 3.1, está dividida em cliente e servidor. O cliente é responsável pela aquisição e visualização do sinal do acelerómetro no *smartphone* e posterior envio para o servidor. O servidor recebe o sinal adquirido e procede à extração de características e classificação de atividades, com um algoritmo previamente treinado.

A aplicação desenvolvida em Android comunica com um servidor Python através de uma implementação open-source disponibilizada pelo projeto Autobahn [53]. O servidor estará à escuta num definido IP e Porto que a aplicação cliente envie o sinal do acelerómetro, que acabou de adquirir.

O bloco responsável pela comunicação em *fullduplex* entre o cliente e o servidor é o WebSocket. As mensagens trocadas podem ser enviadas/recebidas vindas do cliente/servidor.

### 3.1.3 Troca de Informação

De forma a permitir uma troca fácil de informação entre os blocos funcionais foi adoptado o JavaScript Object Notation (JSON). O JSON é um formato leve de troca de dados de fácil utilização homem-máquina, tornando simples para o *developer* interpretar e escrever e para a máquina automaticamente fazer a análise e gerá-lo.

Embora seja baseado num subconjunto da linguagem de programação JavaScript, o JSON é um formato de texto que é completamente independente da plataforma e da linguagem utilizada.

O que torna JSON uma boa solução para o intercâmbio de informação é a possibilidade de utilização de uma estrutura chave / valor e esta ter como base convenções familiares aos programadores de C, C + +, C#, Java, JavaScript, Perl, Python, Matlab e muitos outros.

Esta notação é construído em duas estruturas: uma coleção de pares chave / valor (em várias línguas, este é realizado como um objeto, registo, estrutura, dicionário, lista, ou array), ou uma lista ordenada de valores (na maioria das línguas, isto é percebido como um array, vetor, lista ou sequência).

## 3.2 Arquitetura do Sistema

A arquitetura foi desenvolvida em camadas de forma a ser mais simples o desenvolvimento de novos blocos. É usado o modelo *Model-View-Controller* (MVC) em que é isolada a “lógica” da interface do utilizador permitindo desenvolver, editar e testar separadamente cada parte.

A aplicação é construída por um conjunto de blocos que formam uma rede funcional que corre sobre o Workflow Manager (WFM). Esta rede define o comportamento da aplicação, de modo a que cada bloco tenha uma tarefa e um objetivo específico, como a aquisição de dados da origem, processamento de sinal e apresentação dos resultados.

O núcleo central da aplicação é o módulo do Workflow Manager e a sua função é lidar com a instanciação e controlo do sistema previamente elaborado, descrito de acordo com a estrutura definida no JSON.

### 3.2.1 Blocos Funcionais

Os blocos são as partes operacionais do sistema implementado e a sua ligação cria a *framework* utilizada.

Todos os blocos estendem de classes abstratas que incluem métodos que implementam e observam padrões [54]. A comunicação entre os bloco é baseada em eventos chamados de Message Passing Protocol (MPP) usando a classe Message do SO Android.

A *framework* divide os blocos em três tipos: a) os normais – que recebem informação, fazem uma tarefa específica e produzem resultados de saída; b) as fontes – que é um tipo especial de bloco que apenas tem dados de saída; c) e os finais que são blocos que apenas recebem dados.

As informações sobre as ligações e configurações dos blocos que caracterizam o sistema são apresentadas numa estrutura definida de acordo com a Data Processing Language.

### 3.2.2 Linguagem no processamento de informação

Para a instabilização dos blocos realizada pelo WFM é tido como referência um ficheiro específico baseado em JSON chamado de Data processing Language (DPL). A figura seguinte demonstra um exemplo desta estrutura.

```
{"<CLASS:LABEL>":
{"config": {"<PARAMETER>":"<VALUE>"},...},
"in": {"<CHANNEL>":
"<CHANEL_LABEL>"|"<LABEL>/<CHANNEL>"},...},
"out": {"<CHANNEL>":"<CHANNEL_LABEL>"},...}
},
...
"<CLASS:LABEL>":
{"config": {"<PARAMETER>":"<VALUE>"},...},
"in": {"<CHANNEL>":
"<CHANEL_LABEL>"|"<LABEL>/<CHANNEL>"},...},
"out":{"<CHANNEL>": "<CHANNEL_LABEL>"},...}
} }
```

Este sistema próprio de mensagens facilita a troca de mensagens. O sistema pode ser visto como uma coleção de um par de Chave/Valor no qual são identificados os blocos que a aplicação vai usar e a configuração básica que lhe é dada.

Desde que exista uma correspondência entre os blocos (CLASS em “< CLASS: LABEL”) e classes em Java (CLASS.class, em java), o WFM consegue instanciar e construir o sistema definido quando está em execução. Esta estrutura permite que não seja apenas fácil ler mas também editar.

### 3.2.3 Bloco Acelerómetro

O bloco acelerómetro implementa o “SensorEventListener” de modo a poder aceder aos sensores disponíveis no dispositivo. Como a aplicação se centra num sensor específico, o acelerómetro, é posteriormente realizado o registo de um *listener* para esse sensor. Esse *listener* implementa o método “onSensorChanged()” que é chamado sempre que um novo evento ocorra no acelerómetro.

O *SensorManager* permite o acesso ao sensor do aparelho. Dá uma instancia da classe chamando o *Context.getSystemService()* e o argumento *SENSOR\_SERVICE*.

A classe *SensorEvent* representa um evento do sensor e guarda a informação do tipo de sensor, *timestamp*, precisão e valores obtidos pelo sensor. Quando selecionado o “Sensor.TYPE\_ACCELEROMETER”, como referido no capítulo 2, todas as unidades estão em ( $m/s^2$ ) sendo os valores[0]: aceleração no eixo dos X; valores[1]: aceleração no eixo dos Y; valores[2]: aceleração no eixo dos Z.

O registar do sensor é realizado através do comando “registerListener( SensorEventListener, Sensor, rate)”, onde em *SensorEventListener* é o objecto do sensor que vai estar à escuta, em *Sensor* é escolhido o sensor que se vai registar e em *rate* é especificada a taxa de transferência do sinal, podendo os eventos ser recebidos rapidamente ou lentamente conforme uma das quatro especificações possíveis: “SENSOR\_DELAY\_NORMAL”, “SENSOR\_DELAY\_UI”, “SENSOR\_DELAY\_GAME” ou “SENSOR\_DELAY\_FASTEST”. Neste caso específico como a quantidade de informação é importante para uma eficiente classificação foi selecionado a opção de adquirir os dados do acelerómetro o mais rápido possível, “SENSOR\_DELAY\_FASTEST”.

O bloco acelerómetro grava os dados da aquisição num ficheiro localizado no cartão SD do *smartphone* com a informação do *timestamp* e dos valores da aceleração dos eixos x, y e z num dado instante de tempo.

Embora a gravação do time-stamp e do valor dos três eixos seja gravada no cartão SD do *smartphone* apenas é enviado pelo sistema de blocos o módulo do sinal do acelerómetro, ver equação 4.3, de forma a não sobrecarregar o sistema com demasiada informação.

Com as seguintes linhas apresentadas de seguida é exemplificado como foi realizado o registo acelerómetro dentro do bloco:

```
> mSensorManager = (SensorManager) ctx.getSystemService  
(Context.SENSOR_SERVICE);  
> mAccelerometer = mSensorManager.getDefaultSensor  
(Sensor.TYPE_ACCELEROMETER);  
> mSensorManager.registerListener (this, mAccelerometer,  
SensorManager.SENSOR_DELAY_FASTEST);
```

### 3.2.4 Workflow Manager

O WFM é o núcleo da aplicação e faz a gestão da execução e interação entre os diferentes blocos. Neste bloco é contido um apontador para todas as instâncias importantes do sistema e é através dele que a maioria dos outros blocos comunica.

A tarefa mais exigente realizada pelo WFM é a construção da rede inicial, mas assim que a aplicação começa, a estrutura JSON alimenta o construtor do WFM. É utilizada a biblioteca de alta performance JSON, onde todos os blocos são instanciados sequencialmente, configurados e conectados entre eles através dos seus canais de input e output.

Uma vez o sistema construído, o que resta realizar é iniciar as fontes. Assim que a fonte começa a produzir outputs, que são inputs para os outros blocos, o



sistema flui automaticamente. Tipicamente, a fonte que envia um conjunto de dados notifica todos os blocos registrados para estarem à escuta no canal onde se está a enviar os dados. De seguida, a recepção de dados de um bloco é passada para o input dos próximos blocos, e por ai adiante. Finalmente, o processamento dos dados chega ao final da cadeia, que consiste na apresentação dos dados do acelerómetro ao longo do tempo, num gráfico de linhas no dispositivo.

Normalmente, a ação de iniciar as fonte, é despoletada por um evento UI. Uma vez construída em HTML a interface do utilizador a parte nativa da aplicação deve ser capaz de trocar eventos com o Javascript, utilizando um método chamado de *addJavascriptInterface*, que liga um objecto java a um de javascript. Este objecto está entre a interface Java nativa do Android e Javascript na página HTML. O WFM comunica com a interface chamada de JSI através de mensagens que contêm comandos para ser executados. Por esta razão, o WFM tem também a importante tarefa de analisar e executar estas mensagens de comando. Esse motivo leva a que a única maneira de o JSI comunicar seja através do WFM, que recebe toda a informação e a encaminha corretamente.

### 3.2.5 Interface do utilizador

A interface da aplicação é uma das partes mais importantes da plataforma e necessita de mostrar a informação corretamente mas também de responder aos inputs dados pelos seus utilizadores de forma intuitiva [55].

Outra preocupação tida em conta no design da aplicação foi a unificação do *layout* entre as várias plataformas. Isto leva ao uso da componente Android OS WebView, que permite a interação e exibição do conteúdo HTML, que possibilita o uso de exatamente a mesma interface num *smartphone* ou numa interface standard de PC.

O *layer* de apresentação é desacoplado do processamento *backbone*. Num ambiente móvel, o WFM assegura o processamento em segundo plano, que fornece a informação a ser exibida na interface; no caso o cliente Web, esta tarefa pode ser

realizada por outro *backend* com que a interface do utilizador pode comunicar via websocket. Em qualquer caso, esta comunicação é feita através de um MPP em que a interface do utilizador envia mensagens de comando para acionar ações de *backend*, e vice-versa.

No caso do Android, a interface javascript é configurada para receber comandos do JSI; os clientes que estão a correr o browser, são configurados para receber comandos através do WebSocket. O mecanismo atrás da versatilidade está escondido no script contido no ficheiro HTML. As linhas de código do exemplo DPL demonstradas em cima, são responsáveis pelo comportamento da interface no Android, onde todos os comandos são executados pelo JavaScript que está ligado ao objecto `window.ws`, apenas presente no Android.

Embora não seja perceptível para o utilizador a abordagem realizada é diferente das aplicações usuais das plataformas Android. Usando uma única instância Webview, bem como a abordagem de *layout* unificado, causou a necessidade de um tipo genérico de interface entre o Java e mundo do JavaScript. Assim, JSI é um simples objeto, contendo dois métodos principais: um para o envio de comandos de Java para Javascript e outro para fazer o inverso. De forma a não sobrecarregar o Webkit, responsável por render a página de HTML com o layout, todo o processamento é realizado no Java, antes de chegar ao Webview, porque os métodos nativos do Android são mais eficientes e rápidos que os scripts web.

No decorrer deste projeto foram desenvolvidas duas versões da aplicação. A primeira versão tinha como objetivo construir o *dataset* de treino, na qual o utilizador depois de realizar a atividade a classificava, consoante uma lista atividade predefinidas que lhe era mostrada no ecrã, ver figura 3.2. Na segunda versão da aplicação, o algoritmo já foi treinado para realizar a classificação de atividade e o utilizador já não tem a necessidade de dizer qual a atividade que está a realizar, visto isso ser calculado automaticamente no servidor. Na interface gráfica apenas é demonstrado um gráfico que representa a oscilação do módulo do sinal do acelerómetro, ver equação 4.3. A escolha de apenas representar o módulo do sinal foi diminuir o processamento no *smartphone*.

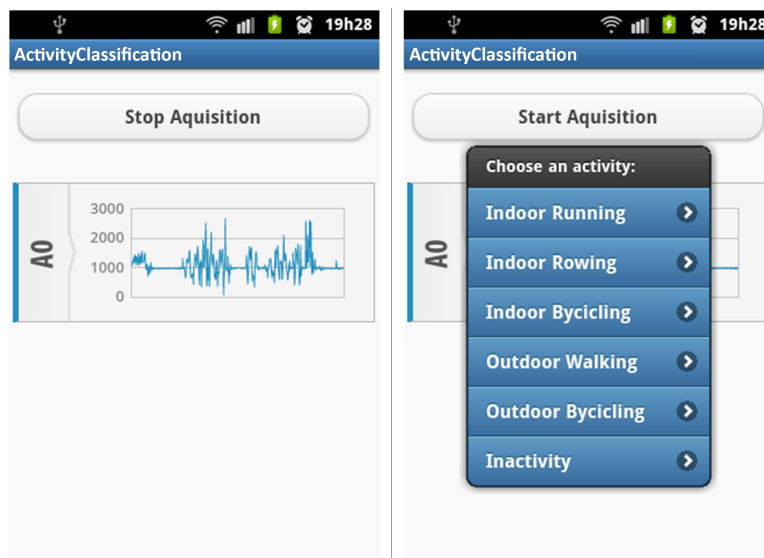


FIGURA 3.2: Captura de ecrã da aplicação realizada neste projeto

### 3.2.6 Comunicação Cliente-Servidor

A comunicação entre cliente e servidor foi efetuada com o uso de bibliotecas open-source que utilizam o protocolo WebSocket disponibilizado pelo projeto Autobahn [53].

Para a comunicação do cliente Android com o servidor em Python foi utilizado a implementação AutobahnAndroid [56]. Do lado do servidor foi utilizado a implementação AutobahnPython [57].

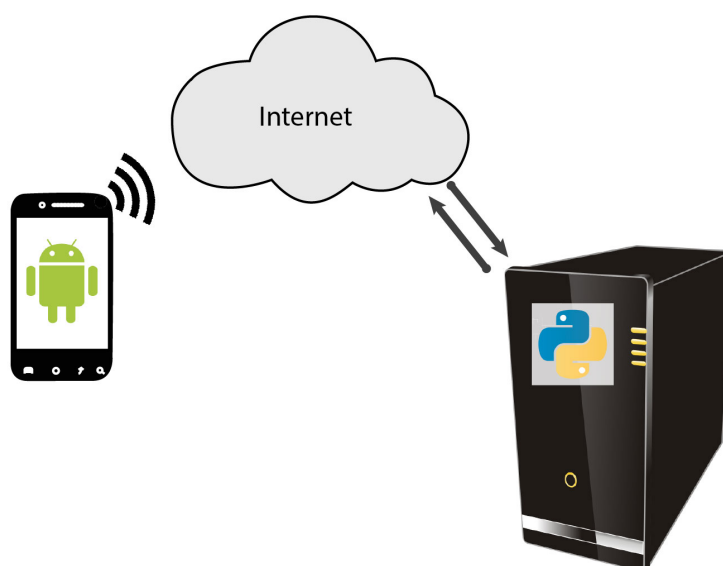


FIGURA 3.3: Comunicação Cliente/Servidor.

O AutobahnAndroid é uma biblioteca open-source de comunicação para Java/Android criada pelo projeto Autobahn que implementa um protocolo WebSocket cliente nativo para *mobile*. Esta ferramenta permite uma ligação rápida às atividades da interface do utilizador assim como aos serviços que estão a correr em *background*.

Para o servidor foi utilizado o AutobahnPython que é uma biblioteca de comunicação para o Python/Twisted. Com esta biblioteca é utilizada uma *framework* que implementa um WebSocket que é escalável e flexível.

O protocolo WebSocket [58] utilizado pela *framework* permite a comunicação bidirecional entre o cliente, que corre um código que não é de confiança num ambiente controlado, para um host remoto que tem como opção comunicação com esse código. O modelo de segurança utilizado para esse efeito é baseado na origem que é usualmente usado pelos *browser* Web.

O protocolo consiste no estabelecimento de um *handshake* (figura 3.4) para abrir a ligação seguido de um enquadramento básico da mensagem, em camadas sobre TCP. O objetivo desta tecnologia é fornecer um mecanismo para aplicativos baseados em *browsers* que necessitam de uma comunicação bidirecional com os servidores que não necessitam de abrir múltiplas conexões HTTP.

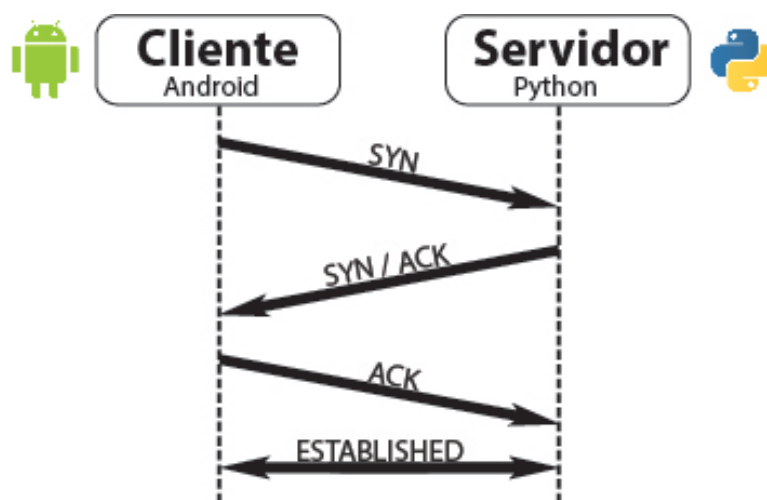


FIGURA 3.4: Figura esquemática da ligação através do protocolo *Handshake*.

## Capítulo 4

# Classificação automática de atividade

A classificação automática de atividade é baseado num sistema de reconhecimento de padrões dividido em três módulos: aquisição do sinal, extração de características e classificação da atividade, como demonstrado na figura 4.1. Esta é a arquitetura clássica para um sistema de reconhecimento de padrões sendo usada neste estudo uma abordagem supervisionada [5]. Os dados foram adquiridos durante atividades previamente definidas, descritas na figura 4.3.

Na primeira parte, o sinal é adquirido do sensor do *smartphone*. Na segunda parte, extração de características, são extraídas algumas características com a finalidade de perceber quais são as características mais distintas de maneira a fazer uma classificação mais eficiente. Na última parte, tentamos usar vários métodos de classificação de maneira a tentar perceber qual é o melhor método para classificar as atividades com as características adquiridas. Serão ainda abordados processos de mineração de dados assim como será efetuada uma contextualização das técnicas de Aprendizagem Automática aplicadas ao problema de classificação de atividades, sendo apresentada a forma de validação utilizada.

Numa primeira abordagem foi construída uma aplicação onde o utilizador classifica a atividade que acabou de fazer, quando premido o botão para parar a aquisição

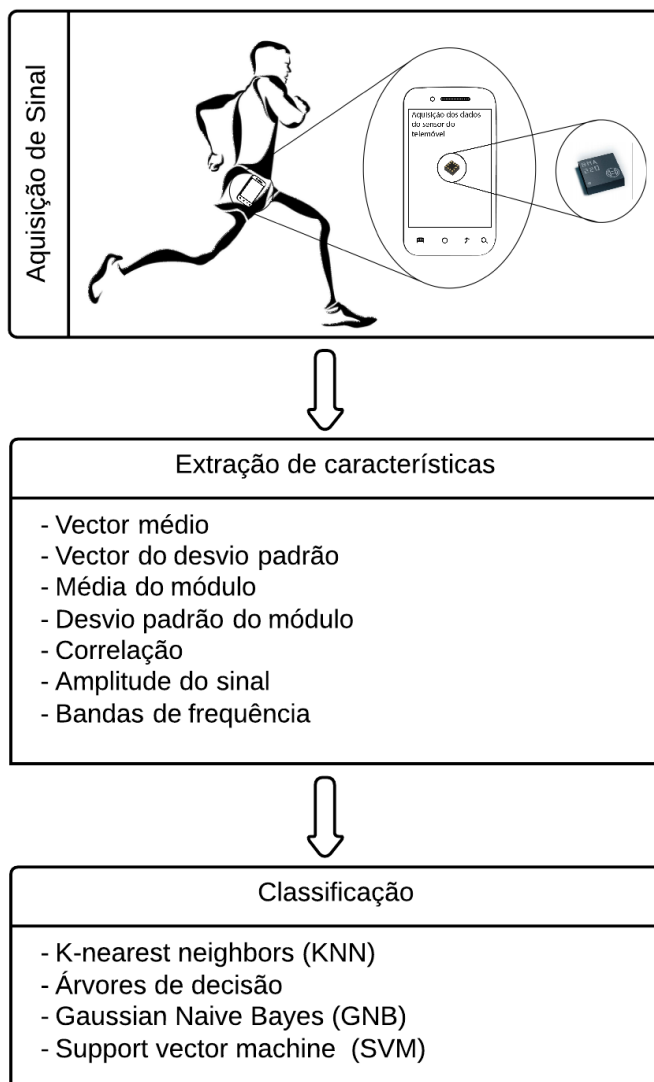


FIGURA 4.1: Arquitetura do sistema proposto

do sinal. Com essa funcionalidade é construído o *dataset* de treino no qual se vai inferir sobre quais as melhores características e algoritmos de classificação, ver figura 3.2.

Na segunda abordagem, já foi escolhido o melhor algoritmo segundo os resultados obtidos pelo *dataset* de treino. Quando o sinal do acelerómetro é adquirido pelo *smartphone* é enviado via *websocket* para um servidor externo onde é realizado a extração de características e classificado com o algoritmo previamente treinado. Depois da classificação é retornado ao utilizador do *smartphone* o resultado da classificação.

## 4.1 Aquisição do sinal

Como já foi referido no Capítulo 2 o modelo do *smartphone* no qual foi realizado o trabalho de desenvolvimento da aplicação foi GT-S5660 (Samsung Galaxy Gio) que corre a versão do Android 2.3.6 e o acelerómetro é fabricado pela BOSCH, modelo BMA 220 [59].

O BMA220 é um acelerómetro de três eixos ultra pequeno, visando aplicações de baixo consumo. O BMA220 permite a medição de acelerações em 3 eixos perpendiculares, e por isso, consegue medir inclinações, movimento, choque e vibrações nos *smartphones*, recursos de realidade virtual e controladores de jogos, entre outros. A seguinte tabela 4.1, mostra as especificações do acelerómetro usado.

Parâmetros	Informação técnica
Digital Resolution	6 bit
Resolution ( $\pm 2$ g range)	62.5 mg
Measurement range (programmable)	$\pm 2$ g, $\pm 4$ g, $\pm 8$ g, $\pm 16$ g
Sensitivity (calibrated)	$\pm 2g : 16LSB/g$ ; $\pm 4g : 8LSB/g$ $\pm 8g : 4LSB/g$ ; $\pm 16g : 2LSB/g$
Bandwidths (programmable)	1 kHz ... 32 Hz

TABELA 4.1: Especificações do modelo de acelerómetro BMA220

No decorrer deste trabalho foi desenvolvida uma aplicação, descrita em pormenor no capítulo 3, que realiza a aquisição do sinal do acelerómetro do *smartphone*. Nessa aplicação desenvolvida para o SO Android, usou-se rotinas dedicadas de acesso ao sensor através do “SensorManager”, classe que permite o acesso aos sensores do dispositivo. Obteve-se uma instância dessa classe, chamando o ”Context.getSystemService()”, com o argumento ”SENSOR\_SERVICE”. Como referido, a aplicação usa como constante ”SENSOR\_DELAY\_FASTEST” de modo a adquirir a informação dos sensores o mais rápido possível. A frequência de amostragem não é constante e interrupta, mas no *smartphone* usado a variação não é significativa, de aproximadamente 95Hz.

Durante a aquisição, o *smartphone* foi colocado dentro do bolso direito de um indivíduo, virado para cima, como é demonstrado na figura 4.1.

O local escolhida para o posicionamento do dispositivo não foi casual, devido ao sinal apresentar nesse local uma grande variação, promovendo melhores resultados. A posição escolhida é conveniente e natural mesmo quando se pratica atividades físicas.

Na aquisição foi considerada a variação intra-sujeito, tendo sido realizadas aquisições em várias sessões.

## 4.2 Segmentação do Sinal

A segmentação do sinal consiste na separação do sinal adquirido em partes mais pequenas, de forma a que cada amostra tenha a duração de 1 minuto de atividade.

A segmentação do sinal é realizada de forma a tentar eliminar inconsistências e redundâncias dos dados de maneira a tornar o *dataset* mais consistente com o universo em estudo. Desta forma tenta-se remover eventuais ambiguidades entre as várias atividades para melhorar o processo de classificação.

Os dados foram divididos em blocos de um minuto onde se removeu o início o fim de cada atividade de forma a tornar o *dataset* mais representativo da atividade praticada. Dessa forma, foi constituído o *dataset* final contendo 50 amostras de cada atividade.

## 4.3 Extração de caraterísticas

Para a extração de caraterísticas, foi usada a abordagem proposta em [32]. Para o estudo do sinal foram realizadas as seguintes extrações de caraterísticas:

A matriz adquirida através do sinal do acelerómetro apresenta três vectores, cada um correspondendo a um eixo.



$$A_{m,n} = \begin{bmatrix} a_{x,1} & a_{y,1} & a_{z,1} \\ a_{x,2} & a_{y,2} & a_{z,2} \\ \vdots & \vdots & \vdots \\ a_{x,n} & a_{y,n} & a_{z,n} \end{bmatrix}$$

onde  $a_{x,t}$ ,  $a_{y,t}$ ,  $a_{z,t}$  corresponde à aceleração no tempo dos três eixos do acelerómetro e ( $t = 1, 2, \dots, n$ ) ao número de amostras.

- **1º e 2º momentos estatísticos da aceleração**

O cálculo dos componentes do vector médio é o seguinte:

$$\bar{a}_i = \frac{1}{N} \sum_{t=1}^N a_i(t), \quad (4.1)$$

onde  $i$  representa os eixos ( $i = x, y, z$ ). O cálculo do desvio padrão de cada componente é o seguinte:

$$\sigma_i = \sqrt{\frac{1}{N-1} \sum_{t=1}^N [a_i(t) - \bar{a}_i]^2}, \quad (4.2)$$

obtendo um vector com as características de cada componente.

- **Cálculo do Módulo do vector médio:**

O cálculo do módulo do vector médio de cada componente é calculado da seguinte forma:

$$|a| = \sqrt{a_x^2 + a_y^2 + a_z^2} \quad (4.3)$$

- **Cálculo do Módulo do desvio padrão médio:**

O cálculo do módulo do desvio padrão médio de cada componente é calculado da seguinte forma:

$$|\sigma^i| = \sqrt{(\sigma_x^i)^2 + (\sigma_y^i)^2 + (\sigma_z^i)^2} \quad (4.4)$$

- **Rápida Transformada de Fourier**

O cálculo da Rápida Transformada de Fourier, do inglês Fast Fourier Transform (FFT), que refere a um modo da discreta transformada de Fourier (DFT) pode ser calculada eficientemente, usando simetrias nos termos do cálculo. As simetrias é maior quando 'n' é elevado a 2, e a transformação é mais eficiente para esses tamanhos.

$$X_k = \sum_{n=0}^{N-1} x_n e^{-j \frac{2\pi}{N} kn} \quad (4.5)$$

- **Potência da densidade espectral**

O espectro do sinal do acelerómetro decompõe o conteúdo de um processo estocástico em frequências diferentes e ajuda a identificar periodicidades. A nossa análise centrou-se em bandas de frequência de 1 Hz, representadas como  $f_i$ . A energia de cada banda é dada por:

$$P(f_i) = \sum_{k \in f_i} |X_k|^2 \quad (4.6)$$

A entropia espectral de cada banda é dada por:

$$S(f_i) = \frac{- \sum_{f_i=0}^{f_n} P(f_i) \log(P(f_i))}{\log(N[(f_{i-1}) - f_i])} \quad (4.7)$$

- **Correlação**

A correlação é a medida de relação entre os três eixos. Esta correlação ajuda estabelecer uma relação entre os eixos e a perceber em qual das direções do sinal é apresentada uma maior variação.

$$R = \begin{bmatrix} \rho_{x,x} & \rho_{y,x} & \rho_{z,x} \\ \rho_{x,y} & \rho_{y,y} & \rho_{z,y} \\ \rho_{x,z} & \rho_{y,z} & \rho_{z,z} \end{bmatrix}$$

A matriz de correlação é obtida da seguinte forma:

$$\rho_{ij} = \frac{1}{N-1} \sum_{t=1}^N \frac{[a_i(t) - \bar{a}_i][a_j(t) - \bar{a}_j]}{\sigma_i \sigma_j}, \quad (4.8)$$

$a_i(t)$  ou  $a_j(t)$  representam os valores num dado instante de tempo num eixo particular, o  $\bar{a}_i$  e  $\bar{a}_j$  representa a média de valores de cada eixo e  $\sigma_i$  ou  $\sigma_j$  o desvio padrão dos eixos.

- **Área da magnitude do sinal:** O cálculo da área da magnitude do sinal é calculada da seguinte forma:

$$SMA = \sum_{t=1}^N |a_x(t)| + |a_y(t)| + |a_z(t)| \quad (4.9)$$

Depois de obtidos as características, foi necessário tornar os dados apropriados para a classificação, organizando e normalizando as características.

## 4.4 Classificação

Na classificação é utilizado o método da aprendizagem automática supervisionada [4, 5]. Deste modo pretende-se compreender como criar métodos que permitam melhorar o desempenho das máquinas em determinadas tarefas, nomeadamente através do conceito da "experiência". Dessa forma, os algoritmos têm-se revelado extremamente úteis em diversos domínios, tendo sido especialmente importantes na resolução de problemas de mineração de dados, onde universos de dados extensos poderão conter implicitamente informação considerada relevante para uma classificação eficaz e dessa forma conseguir classificar com elevada precisão.

Neste estudo, foi realizada uma aprendizagem supervisionada e utilizados vários classificadores de modo a assegurar que os resultados obtidos são fiáveis e estatisticamente significativos.

Existem vários métodos para assegurar a avaliação da qualidade e características de um modelo, que incluem nomeadamente a utilização de métricas de validação estatísticas, permitindo detectar possíveis anomalias nos dados adquiridos ou no próprio modelo.

Para determinar a precisão dos classificadores é necessário verificar se o modelo está de acordo com os resultados obtidos a partir das características extraídas dos dados adquiridos assim como avaliar o modo como o modelo se comporta em conjuntos de dados diferentes. Caso sejam apresentados bons resultados, encontrando bons padrões e prevendo com o menor erro possível, será possível dizer que estamos perante um modelo fiável.

Este estudo explora o uso de classificação paramétrica supervisionada usando algoritmos como o Gaussian Naive Bayes e não-paramétricos como o k-Nearest Neighbor, Árvores de decisão e Support Vector Machines (SVM). Para a classificação foram usados os seguintes métodos [5]: *K-Nearest Neighbors*, com 1 e 3 vizinhos, e com todos os pontos em cada vizinhança com pesos iguais; Gaussian Naive Bayes, com o desvio padrão e a média estimados usando a máxima verosimilhança; Árvores de Decisão em que o objetivo é criar um modelo que prevê o valor de uma variável alvo, aprendendo regras simples decisão inferidas a partir das características recolhidas dos dados; e Linear Support Vector Classification usando regularização com a norma L1 que induz dispersão.

Neste projeto foi utilizado a validação cruzada com K-Fold como método para validar os resultados. Neste método os dados são divididos de forma aleatória em subconjuntos de K elementos de tamanhos iguais, para desse modo ser realizado uma aprendizagem de K iterações. O processo é repetido para as K amostras e a performance do classificador é avaliado de acordo com a média adquirida nos K testes realizados, ver figura 4.2. Este método tem como principal característica utilizar todo o *dataset*.

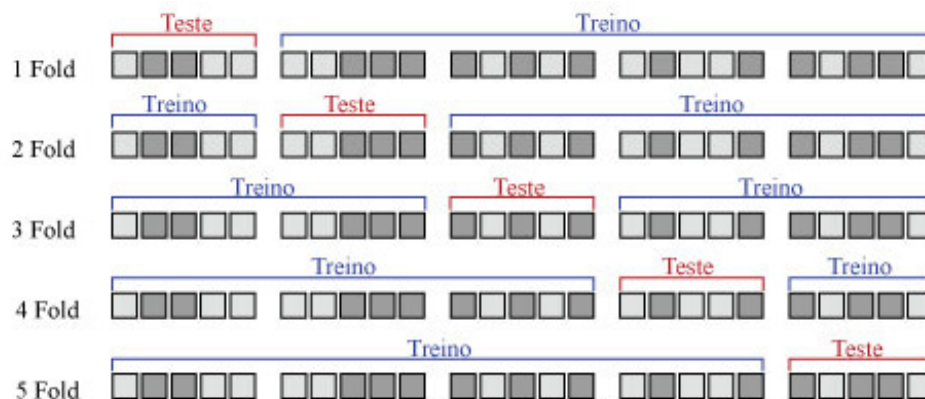


FIGURA 4.2: Representação esquemática do método de validação cruzada com 5-fold

De forma a tentar perceber quais as características mais relevantes para a classificação foi utilizado o modelo linear com a penalização pela norma L1 (distância de Manhattan). Será possível, através dessa informação perceber quais as características mais significativas para a classificação e futuramente reduzir o tamanho do universo de dados, removendo características redundantes ou irrelevantes. Ter-se-á como objetivo, encontrar o menor número de características de forma a que a probabilidade resultante da distribuição das várias classes se aproxime o mais possível da distribuição original, tornando mais compreensível os padrões existentes.

As atividades consideradas neste estudo são: em ambiente *Indoor* (Correr, Remo e Bicicleta) e em ambiente *Outdoor* (Bicicleta e Andar), ver figura 4.3. É importante salientar que neste estudo a inatividade é considerada uma atividade e que as aquisições em ambiente *Indoor* foram efetuadas num ginásio.

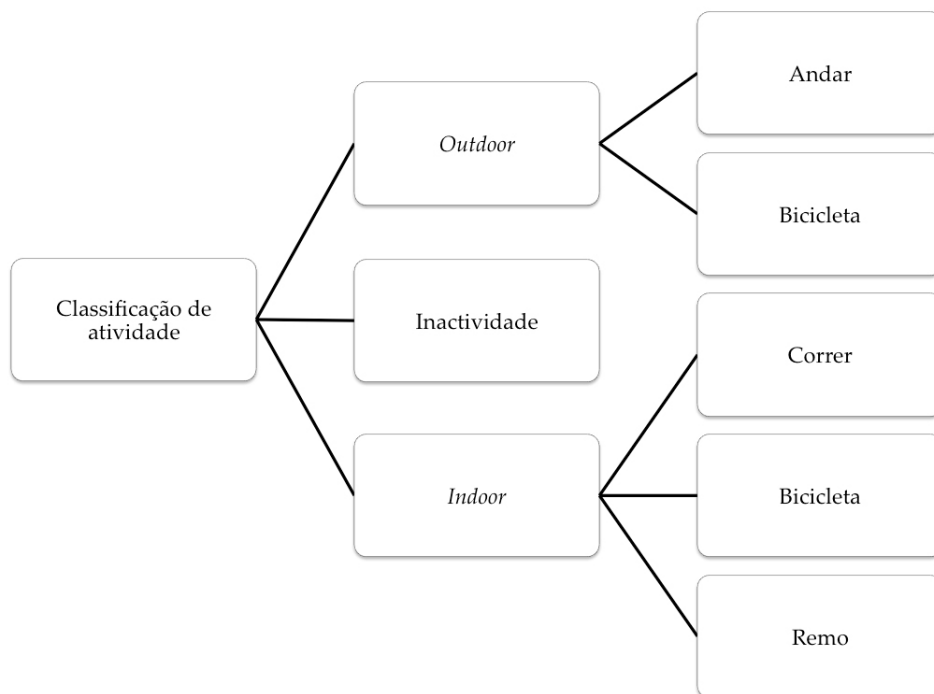


FIGURA 4.3: Atividades consideradas neste estudo

# Capítulo 5

## Estudo da atividade

### Andar/Correr

Este projeto foca-se, sobretudo, na classificação automática de atividades mas como trabalho futuro poderia ser realizado um estudo de cada uma das atividades consideradas neste projeto.

Devido às limitações temporais, foi apenas possível realizar um breve estudo sobre a contagem de passos que poderá ser aplicado às atividades: andar e correr.

#### 5.1 Enquadramento

Os dispositivos que fazem a contagem de passos são chamados de pedómetros. Estes dispositivos estão a ganhar importância nos dias de hoje e a tornar-se numa ferramenta para motivar a prática de atividades físicas. Esta ferramenta possibilita o controlo aproximado das calorias gastas, a medição do progresso ao longo do tempo, o cálculo da distância percorrida e da velocidade, entre outros aspetos.

Os pedómetros realizam a contagem de passos e para o fazer é necessário ter em atenção o ciclo de andamento do ser humano. Esse ciclo de andamento divide-se em quatro tarefas: início e término dos movimentos locomotores, a criação de

movimento contínuo para o progresso numa determinada direção, a manutenção do equilíbrio e a capacidade de adaptação às mudanças ambientais [60].

Caminhar é uma atividade cíclica em que um passo segue outro num padrão contínuo, ver figura 5.1. Segundo o livro *Introduction to Sport Biometrics* [35], uma passada é definida desde o levantar de um pé do chão até o pouso desse mesmo pé, ou do levantamento dos dedos até o pouso dos mesmos. Existem duas fases de posicionamento dos pés; são essas: fase de posicionamento único e duplo, sendo que a primeira acontece quando apenas um pé está assente no chão e a segunda quando ambos os pés estão assentes no chão. A primeira fase, inicia-se com o levantamento de um pé do chão, caracterizando o início de uma passada, até ao assentar desse mesmo pé, iniciando a segunda fase onde existe duplo apoio no chão. A duração da fase de suporte único é cerca de quatro vezes maior do que a fase de suporte duplo.

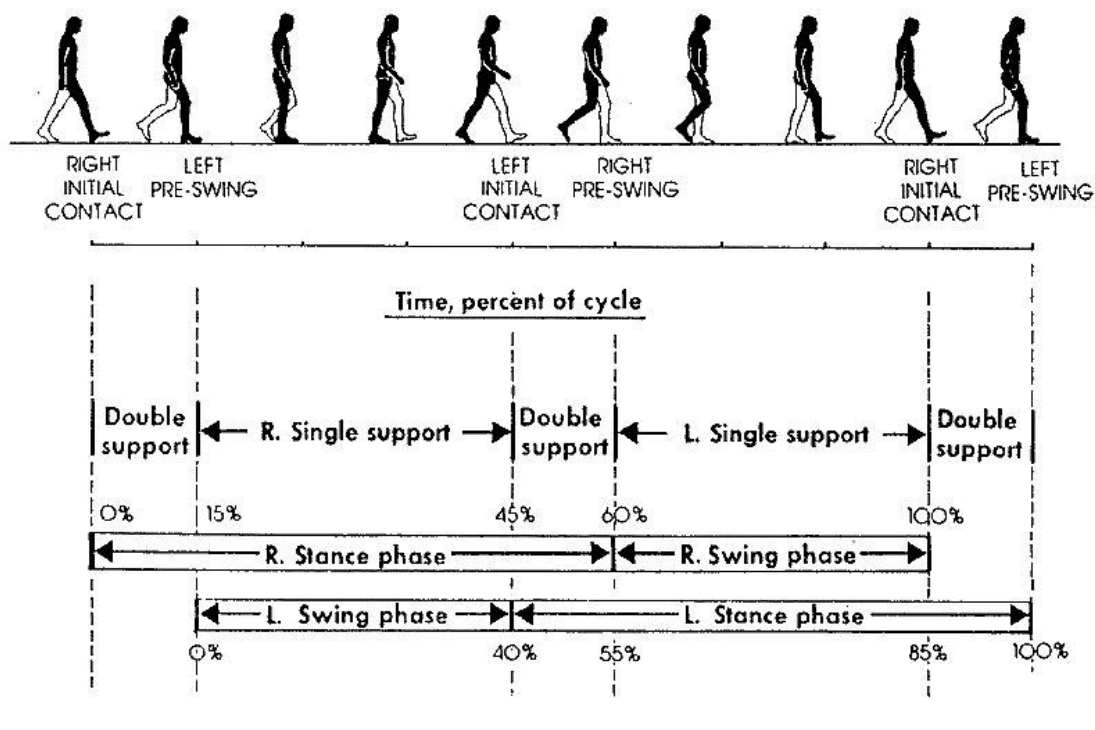


FIGURA 5.1: Ciclo de andamento (Fonte: UT Dallas [61])



## 5.2 Estudo exploratório

Neste projeto foi realizado um estudo exploratório sobre a contagem de passos através do sinal adquirido pelo acelerómetro. Após o enquadramento sobre o ciclo de andamento humano e olhando para o sinal adquirido pelo *smartphone*, foi desenvolvido um algoritmo para proceder à contagem de passos.

Segundo a literatura, existem sobretudo três métodos de deteção de um passo, são esses [62]:

- *Peak Detection*: Procura as maiores variações no sinal. As variações serão descartadas caso sejam inferiores ao limiar definido.
- *Zero crossing detection*: A função “zero-crossing” é um ponto onde o sinal da função muda, representado pelo cruzamento do eixo (valor zero) no gráfico da função.
- *Flat zone detection* usando um diferencial de aceleração: Procura zonas planas no sinal.

Algumas das técnicas utilizadas para validação de um passo com maior precisão são[29]:

- Utilização de um filtro digital: Método para transformação do sinal tornando-o mais regular. Essa transformação poderá ser realizada calculando o módulo do sinal, removendo as dependências da orientação e retirando a componente gravitacional do sinal.
- Limiares dinâmicos e precisão dinâmica: Uso de limiares e precisão dinâmicos para determinar quando o passo é realizado, verificando continuamente os valores máximos e mínimos dos três eixos de aceleração num número específico de amostras. Recorre-se aos limiares dinâmicos para definir com mais precisão se o passo foi efetuado ou não.

- Detecção do picos: Cálculo do número de passos através de um algoritmo de detecção de picos, verificando os eixos que apresentam as maiores variações. As que apresentarem variações menores serão descartadas.
- *Time Windows*: Utilização de uma janela deslizante para descartar vibrações inválidas.
- Padrão de andamento: Deverá ter-se em atenção se os passos estão dentro do ritmo do andamento padrão.

### 5.3 Informações sobre a atividade praticada

Com base nos três métodos de contagem de passos acima descrito, desenvolveu-se, nesta parte exploratória do trabalho, o algoritmo de detecção de picos, "Peak Detector".

Na figura 5.2 é demonstrada a aplicação do algoritmo desenvolvido num pequeno segmento do sinal adquirido pelo acelerómetro do *smartphone*. O sinal demonstrado apresenta cerca de 10 segundos (1000 amostras) da atividade andar.

Nesse segmento pode-se observar que foram dados cerca de 16 passos, valor padrão para a velocidade de andamento normal. Destacam-se ainda, visualizando os picos superiores assinalados com círculos azuis, as duas fases do ciclo de andamento.

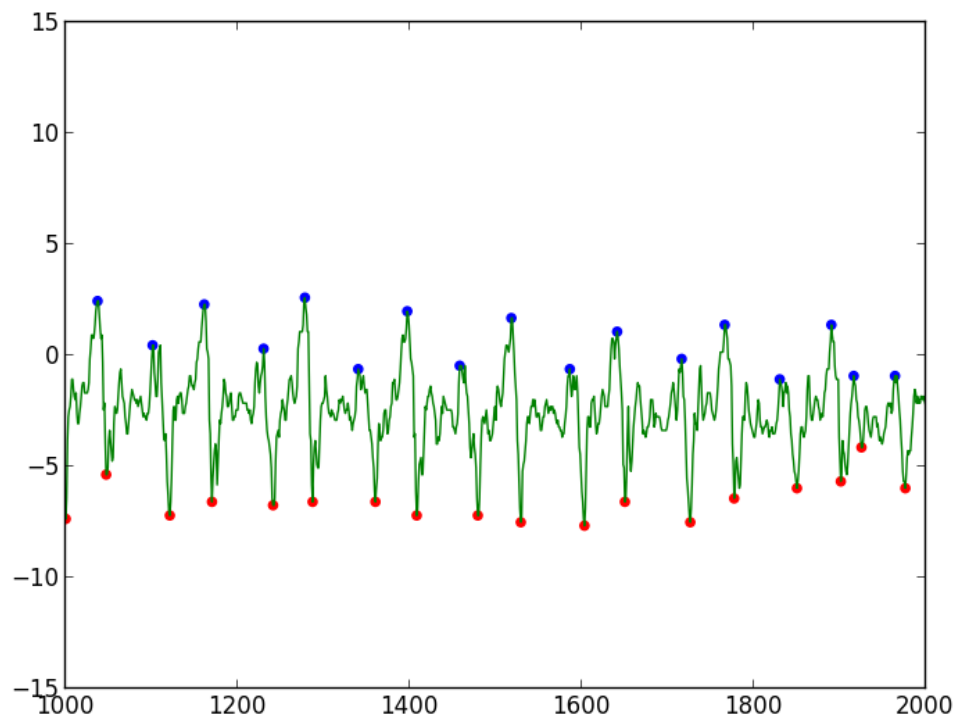


FIGURA 5.2: Algoritmo *Peak Detector* aplicado a um segmento do sinal adquirido pelo *smartphone*

De forma a obter mais informação sobre a atividade praticada são apresentadas de seguida, de forma simplificada e aproximada, o cálculo da distância, velocidade e da energia gasta [29].

- **Cálculo do comprimento do passo:** No caso de se possuir a informação do sexo do indivíduo que pratica a atividade, é possível calcular o comprimento estimado do passo, multiplicando a sua altura por uma constante que varia consoante o sexo.

Sexo Masculino:

$$\text{Comprimento da Passo (m)} = \text{Altura (m)} \times 0.415 \quad (5.1)$$

Sexo Feminino:

$$\text{Comprimento da Passo (m)} = \text{Altura (m)} \times 0.413 \quad (5.2)$$

- **Cálculo da distância:**

O cálculo aproximado da distância é realizado multiplicando o comprimento de cada passo (ver equações 5.1, 5.2) pelo número total de passos dados durante a atividade.

$$\text{Distância Percorrida (m)} = \text{Comprimento do passo (m)} \times \text{Número de passos} \quad (5.3)$$

- **Cálculo da velocidade:** O cálculo aproximado da velocidade é obtido pela seguinte fórmula:

$$\text{Velocidade (m/s)} = \text{Distância (m)} / \text{Duração (s)} \quad (5.4)$$

- **Cálculo da energia gasta:**

Para o cálculo da energia gasta é tido em conta o estudo [63], de onde se retiram os METs de referência em cada atividade (ver tabela 5.1) e se multiplica esse valor por uma constante, pela duração da atividade e pelo peso do indivíduo. O cálculo da energia gasta é realizado através da seguinte fórmula:

$$\text{Calorias Gastas (kcal)} = 1.05 \times \text{METs} \times \text{Duração (h)} \times \text{Peso (Kg)} \quad (5.5)$$

Para as atividades consideradas neste estudo, e usando os dados disponibilizados por [63], apresentam-se na tabela 5.1 os equivalentes metabólicos genéricos correspondentes a cada uma das atividades consideradas.

<b>METS</b>	<b>Atividade</b>	<b>Descrição da Atividade</b>
8	Bicicleta	Bicicleta, genérico
7	Exercício condicionado: Remo	Remo, estacionário, genérico
10	Correr	6 mph (10 min/mile)
1	Inatividade, quieto	Sentado ou deitado, sossegado
7	Exercício condicionado: Bicicleta	Bicicleta, estacionário, genérico
3.3	Andar	Andar, 3.0 mph, área plana, passo moderado, superfície firme

TABELA 5.1: Equivalentes metabólicos genéricos das atividades consideradas neste estudo



# Capítulo 6

## Análise de resultados

Neste capítulo são apresentados os resultados obtidos. A análise desses resultados divide-se em quatro partes. Na primeira são analisadas as particularidades do sinal adquirido; na segunda, estudam-se as diferenças entre as várias características extraídas; na terceira parte, analisam-se os resultados obtidos pelos vários classificadores e procura-se perceber quais as atividades mais confundidas; e na última, é analisada a informação extraída das atividades andar/correr.

Ao longo do capítulo apresentam-se e discutem-se os resultados.

### 6.1 Sinal Adquirido

Na figura 6.1 são representados os sinais do acelerómetro adquiridos durante a execução das várias atividades consideradas. De forma a diminuir a complexidade e a tornar mais perceptível as diferenças na aceleração, as subfiguras apresentam somente o módulo do sinal adquirido em cerca 60 segundos de atividade.

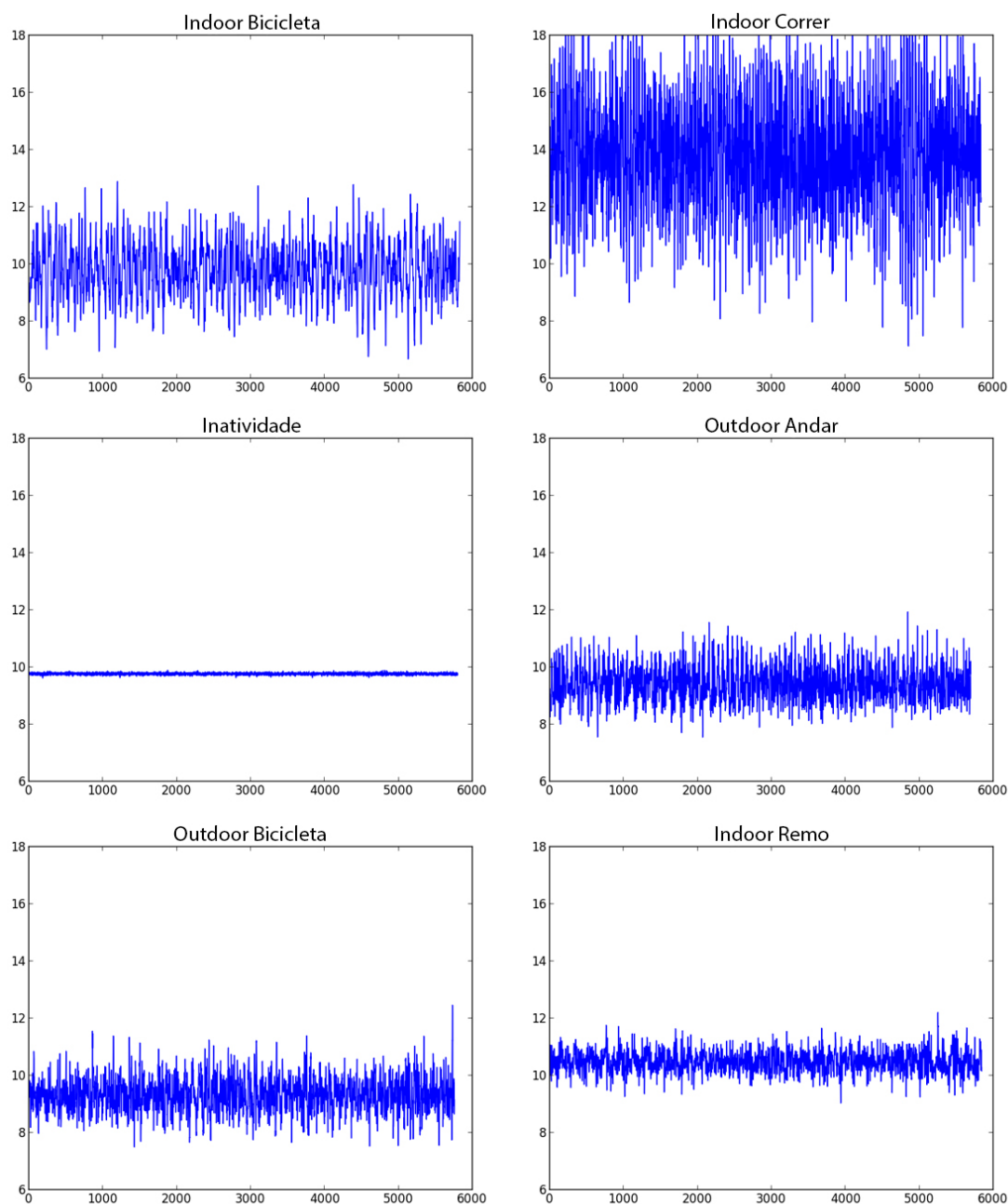


FIGURA 6.1: Módulo do sinal adquirido pelo acelerômetro do smartphone em cada atividade.

Como já referido no capítulo 3, a frequência de amostragem não é constante. Na figura 6.2 é apresentado um histograma da frequência de amostragem obtida durante 60 segundos de atividade. Ao observar a figura é possível verificar que a maioria das amostras foram adquiridas no intervalo 0-0,04, correspondendo a aproximadamente 95Hz. Contudo, como a frequência de amostragem não é constante,



no pior dos casos pode chegar aos 75Hz.

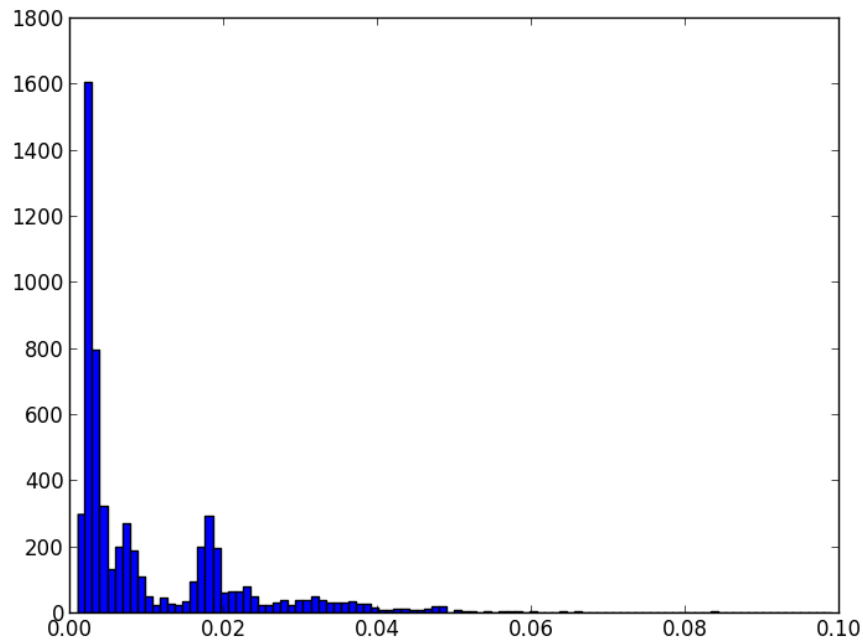


FIGURA 6.2: Histograma representando a frequência de amostragem

## 6.2 Exploração de Características

Para compreender melhor as características no domínio do tempo, apresentadas no capítulo 4, e devido às diferenças de valores, efetua-se a seguinte separação: características estatísticas ( $a_x$ ,  $a_y$ ,  $a_z$ ,  $\sigma_x$ ,  $\sigma_y$ ,  $\sigma_z$ ,  $|a|$ ,  $|\sigma|$ ), SMA e valores da correlação ( $\rho_{x,y}$ ,  $\rho_{x,z}$  e  $\rho_{y,z}$ ).

Nas figuras 6.3, 6.4 e 6.5 cada barra com uma determinada cor representa uma atividade diferente e mostra o valor médio dessa característica, assim como o seu desvio padrão, valores esses obtidos com base nas várias aquisições de sinal realizadas.

Na figura 6.3 são apresentadas as características estatísticas, tais como médias e desvios padrões. Observando a figura consegue-se distinguir claramente as atividades *Indoor* Correr e Inatividade. A primeira atividade apresenta valores elevados de  $\sigma_x$ ,  $\sigma_y$  e  $\sigma_z$ , enquanto que a Inatividade apresenta valores baixos nessas três

características. As atividades *Indoor* Bicicleta e *Outdoor* Bicicleta apresentam valores semelhantes em todas as características do domínio do tempo apresentadas na figura.

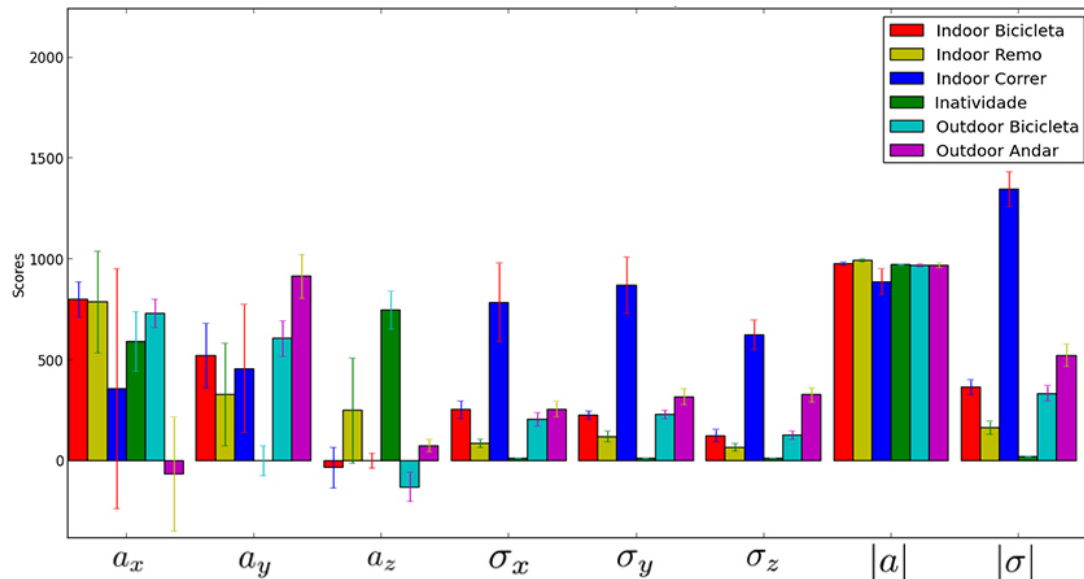


FIGURA 6.3: Média e desvio padrão de um conjunto de características estatísticas no domínio do tempo.

Na figura 6.4 são apresentados os valores da característica SMA. Esta característica não parece ser a mais indicada para distinguir as várias atividades pelo facto de os valores obtidos serem idênticos. De destacar é a semelhança nos valores das atividades *Indoor* Bicicleta com *Outdoor* Bicicleta e o elevado valor do desvio padrão das atividades *Indoor* Correr e *Outdoor* Andar.

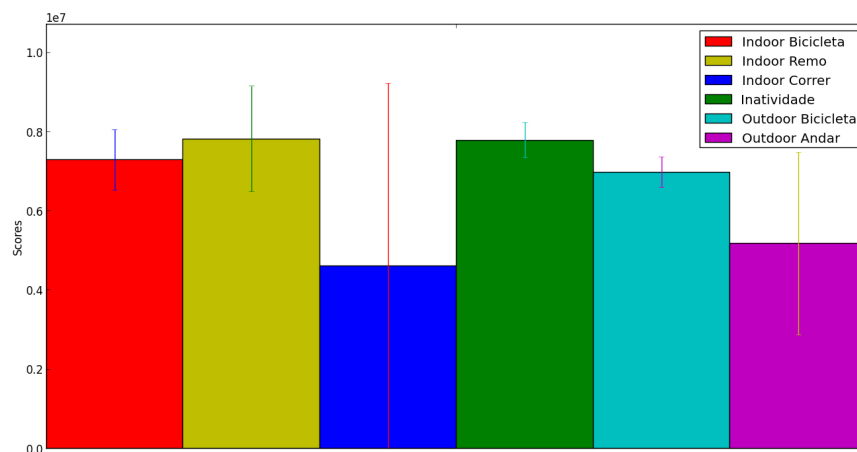


FIGURA 6.4: Média e desvio padrão da característica SMA de cada atividade.

Ainda no domínio do tempo, na figura 6.5 estão representados os valores da diagonal superior da matriz de correlação. De realçar nessa figura o elevado valor na componente  $\rho_{y,x}$  na atividade *Indoor Correr* ou o elevado desvio padrão da componente  $\rho_{z,y}$  da atividade *Inatividade*.

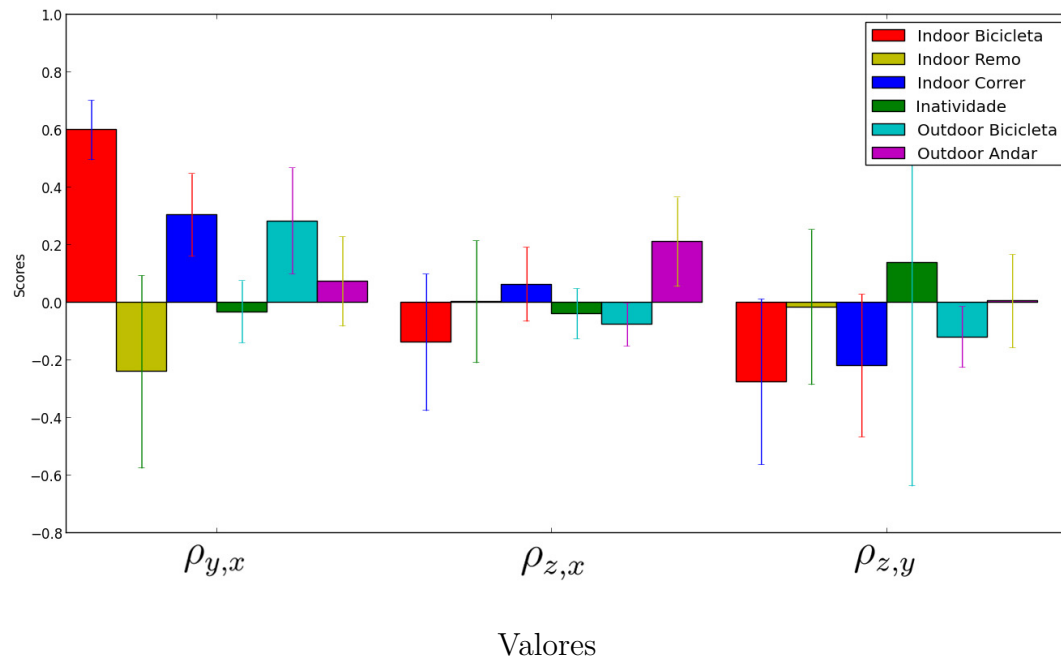
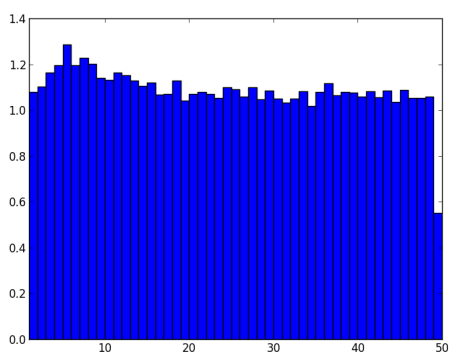


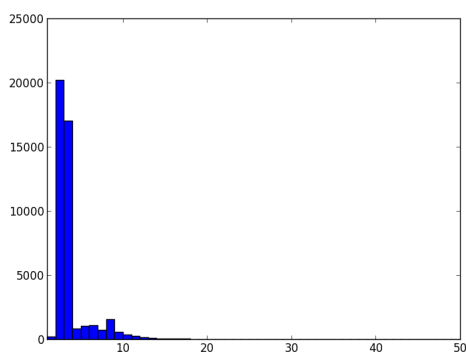
FIGURA 6.5: Média e desvio padrão dos três elementos de correlação de cada atividade.

No domínio da frequência são estudadas as bandas de potência em intervalos de 1Hz.

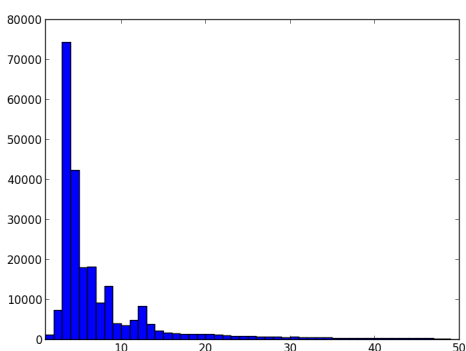
Numa primeira análise é efetuado um gráfico de barras representando as primeiras 50 bandas de potência de cada atividade, figura 6.6. As escalas apresentadas em cada atividade são diferentes, tendo como objetivo perceber a sua distribuição.



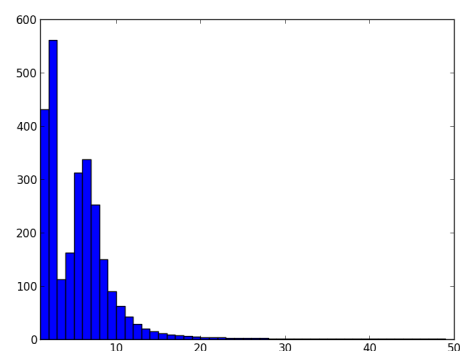
(a) Inatividade.



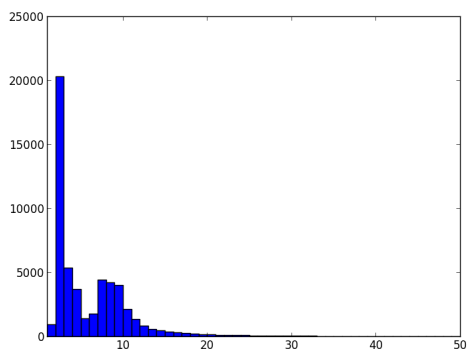
(b) Indoor Bicicleta.



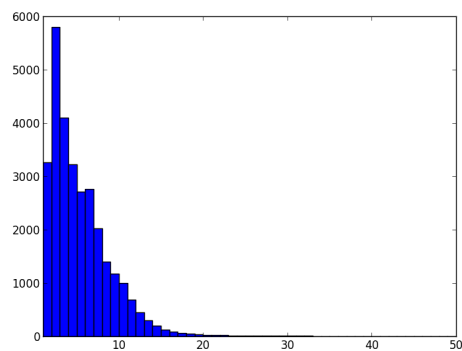
(c) Indoor Correr.



(d) Indoor Remo.



(e) Outdoor Andar.



(f) Outdoor Bicicleta.

FIGURA 6.6: Distribuição das bandas de potência de cada atividade (0-50Hz)

De modo a conseguir comparar os valores das bandas de potência obtidos nas várias atividades foi elaborada a figura 6.7, onde se apresenta a média das trinta

primeiras bandas de potência de cada atividade. Observando a figura, conseguem-se distinguir algumas atividades, tais como a *Outdoor* Bicicleta de *Indoor* Remo ou *Indoor* Correr da Inatividade.

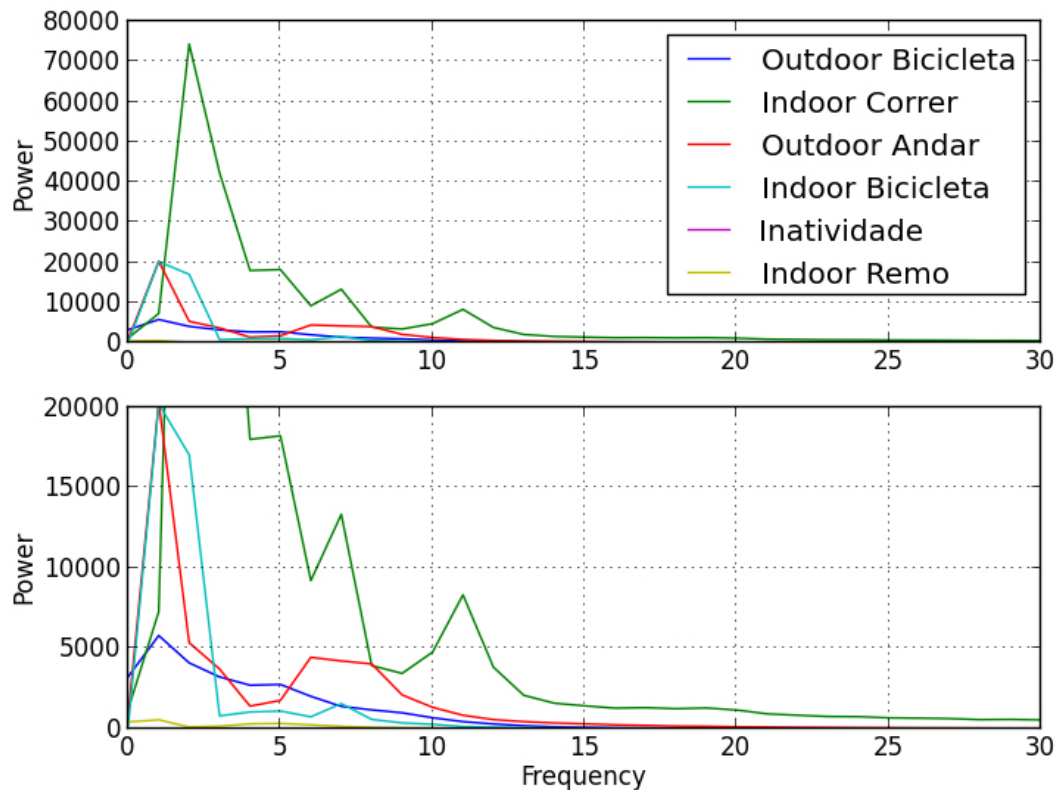


FIGURA 6.7: Valores médios das bandas de frequência de cada atividade.

### 6.3 Resultados da Classificação

Os algoritmos de classificação foram desenvolvidos com o recurso a python 2.7 baseado no scikit learn [64]. Este módulo é uma biblioteca *open source* de aprendizagem automática, do inglês *machine learning*, para a linguagem de programação Python. O módulo possui ferramentas disponíveis para classificação, regressão, *clustering* e foi desenvolvido para trabalhar com bibliotecas numéricas e científicas, tais como NumPy e SciPy, respetivamente.

Para evitar o *over-fitting*, definimos dois diferentes conjuntos de dados: o *dataset* de treino  $X^l, y^l$ , que é usado para aprender a função de previsão e o *dataset* de

teste  $X^t$ ,  $y^t$ , que é usado para testar a função de previsão. Contudo, definindo estes dois conjuntos, reduz-se drasticamente o número de amostras que podem ser usadas para o modelo de aprendizagem, e os resultados podem depender de um par particular de conjuntos de treino e teste.

A solução foi separar todo o *dataset* várias vezes consecutivas em diferentes *datasets* de treino e teste e calcular a média de valores retornados da previsão, obtidos com os diferentes conjuntos. Como esta abordagem pode ser computacionalmente exigente, é realizada no servidor e apresenta como vantagem não desperdiçar dados (como era o caso se fosse realizada a fixação de um conjunto de teste arbitrário), importante em problemas onde o número de amostras é reduzido.

O método utilizado para treinar cada classificador foi a validação cruzada, usando uma estratégia K-Fold. O método de validação cruzada denominado por K-Fold divide o *dataset* em  $k$  subconjuntos mutualmente exclusivos do mesmo tamanho. Dessa forma, um subconjunto é utilizado para teste e os restantes  $k - 1$  são utilizados para estimar os parâmetros e calcular o modelo através da média dos resultados. Este processo é realizado  $k = 5$  vezes alternando, de forma circular, o subconjunto de teste. De cada uma das 5 vezes que é realizado, obtem-se um determinado resultado que, no final, servirá para calcular a média e o desvio padrão de cada classificador.

Foram calculados os resultados obtidos pelos vários classificadores no domínio do tempo, da frequência e de ambos.

Os resultados baseados no domínio do tempo, com as seguintes características – a média e o desvio padrão dos três eixos, média e o desvio padrão do módulo, a SMA e os valores da diagonal superior da matriz de correlação – são descritos na tabela 6.1:

<b>Algoritmos</b>	<b>Precisão</b>
<b>1NN</b>	0.76 (+/- 0.02)
<b>3NN</b>	0.69 (+/- 0.01)
<b>Gaussian Naïve Bayes</b>	0.96 (+/- 0.01)
<b>Decision Tree</b>	0.97 (+/- 0.01)
<b>Linear SVM</b>	0.91 (+/- 0.01)

TABELA 6.1: Resultados da classificação no domínio do tempo.

Os resultados baseados nos valores do domínio da frequência são apresentados na tabela seguinte:

<b>Algoritmos</b>	<b>Precisão</b>
<b>1NN</b>	1.00 (+/- 0.00)
<b>3NN</b>	1.00 (+/- 0.00)
<b>Gaussian Naïve Bayes</b>	0.98 (+/- 0.01)
<b>Decision Tree</b>	0.98 (+/- 0.00)
<b>Linear SVM</b>	0.96 (+/- 0.01)

TABELA 6.2: Resultados da classificação no domínio da frequência.

Finalmente, tentou-se perceber se, usando as características de ambos os domínios, tempo e frequência, os resultados da classificação melhoravam. Na tabela seguinte apresentam-se os resultados:

Algoritmos	Precisão
1NN	1.00 (+/- 0.00)
3NN	1.00 (+/- 0.00)
Gaussian Naïve Bayes	0.98 (+/- 0.00)
Decision Tree	0.97 (+/- 0.01)
Linear SVM	0.97 (+/- 0.01)

TABELA 6.3: Resultados da classificação no domínio do tempo e da frequência.

De forma a tentar encontrar as características mais importantes recorreu-se ao modelo linear com penalização da norma L1, estimando os coeficientes que não eram zero. As SVMs têm o parâmetro de penalização C que induz a separação dos dados, foram escolhidos valores variando entre  $[1^{-3}, 1^{-2}, \dots, 1^5]$  de forma a separar de forma eficiente os dados. Depois de aplicados os modelos lineares penalizados com a norma L1, as características com coeficientes não zero no domínio do tempo são  $a_z, \sigma_x, \sigma_y, \rho_{y,x}$  e  $\rho_{z,x}$  (usando um C=1); e no domínio da frequência as bandas espectrais de frequência dos 0-10Hz, 11-12Hz, 19-21Hz que representa 14 bandas de um total de 30 consideradas (C=0.001).

A aquisição foi realizada com o *smartphone* numa posição idêntica, levando a bons resultados no domínio do tempo. Espera-se que se houver mudanças extremas nos eixos haja uma perda de eficácia dos classificadores que têm por base características do domínio do tempo.

Nas figuras 6.8 e 6.9 estão representadas as matrizes de confusão realizadas para avaliar a precisão dos classificadores e perceber quais as atividades mais confundidas. A título de exemplo foi selecionado um classificador para as características do domínio do tempo e outro para as características do domínio da frequência.

No domínio do tempo, utilizando o classificador GNB, as atividades mais confundidas são *Indoor* Bicicleta e *Outdoor* Bicicleta, ver figura 6.8.



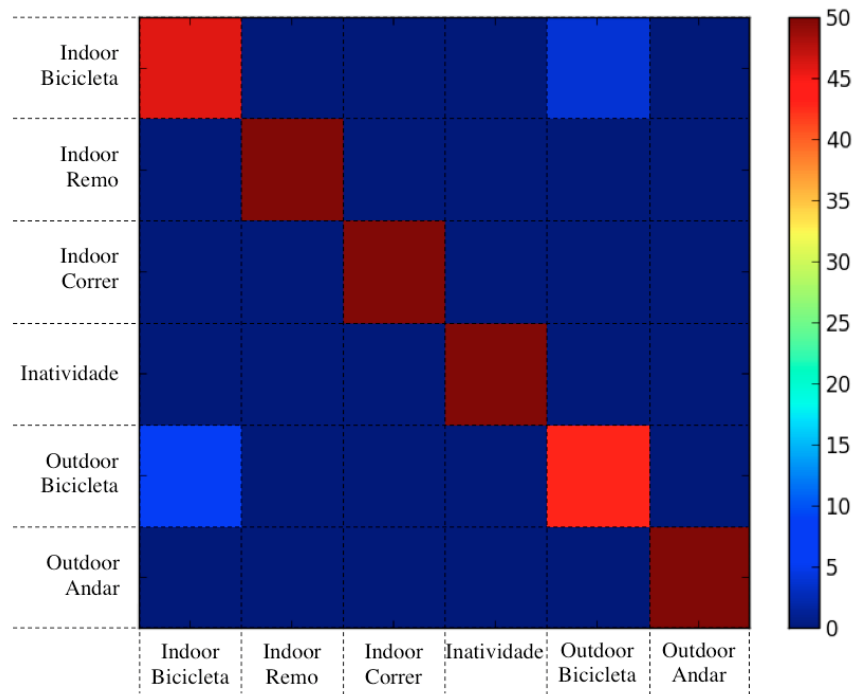


FIGURA 6.8: Matriz de confusão do classificador GNB para as características do domínio do tempo.

No domínio da frequência, utilizando o classificador Árvore de Decisão, as atividades mais confundidas são *Outdoor Bicicleta* com *Outdoor Andar* e *Indoor Bicicleta*, ver figura 6.9.

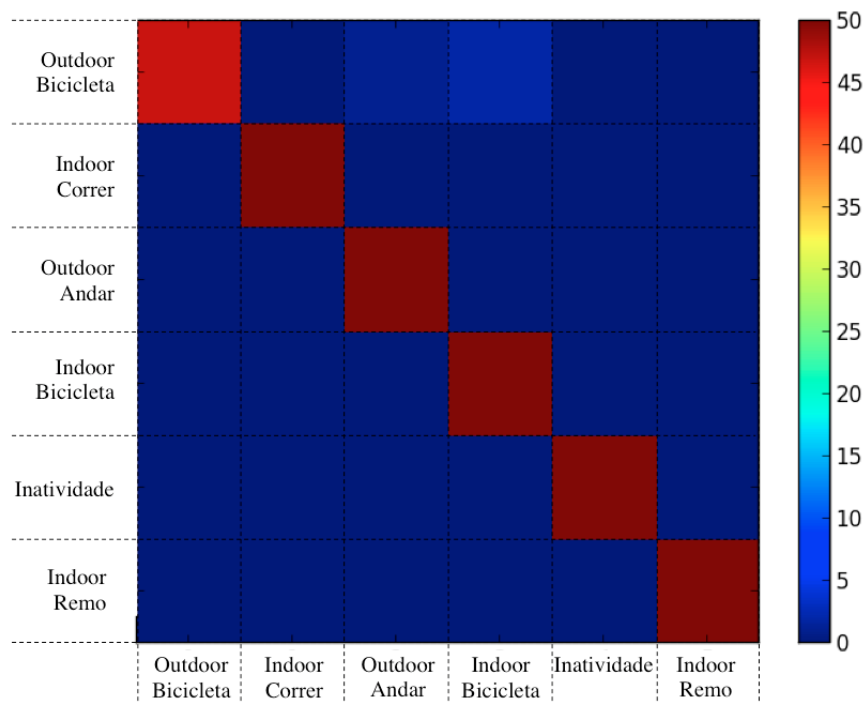


FIGURA 6.9: Matriz de confusão do classificador Árvores de Decisão para as características do domínio da frequência.

## 6.4 Informação sobre a atividade

No capítulo 5 foi efetuado um estudo exploratório do número de passos, distância, velocidade e calorias gastas. Na tabela 6.4 apresentam-se os resultados dos algoritmos, aplicados a cinco segmentos da atividade Andar Outdoor com uma duração de 60 segundos.

De forma a demonstrar o algoritmo de deteção de passos, na figura 6.10 ilustra-se o algoritmo aplicado a um dos segmentos presentes na tabela.

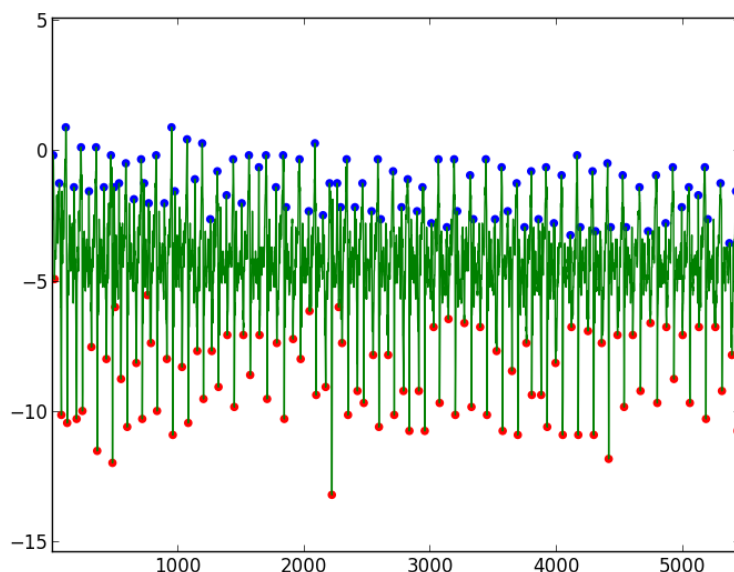


FIGURA 6.10: Algoritmo de detecção de picos aplicado a um segmento de 60 segundos da atividade Andar Outdoor.

Para o cálculo do comprimento do passo foi considerada a equação 5.1, calculada tendo em conta um indivíduo do sexo masculino com uma altura de 1,82m. Segundo esses dados, o comprimento da passada obtido é de 0,7553, valor utilizado posteriormente para calcular a distância e, conseqüentemente, a velocidade apresentada na tabela 6.4.

Passos Efetuados	Passos Detetados	Distância	Velocidade
92	93	70,9982	1,170 m/s
92	94	70,2429	1,183 m/s
92	92	69,4876	1,158 m/s
92	96	72.5088	1,208 m/s

TABELA 6.4: Informação sobre a atividade Andar Outdoor a cinco aquisições de sinal com duração de 60 segundos.

Para o cálculo da energia gasta (ver equação 5.5), considera-se que o indivíduo está a praticar a atividade Andar Outdoor, o valor do equivalente metabólico da atividade Andar Outdoor é de 3.3 (ver tabela 5.1) e que a duração da atividade foi de 60 segundos. Tendo em conta esses valores, a energia gasta é de 3.928 kcal.



# Capítulo 7

## Conclusão e Trabalho Futuro

Neste projeto foram realizadas várias abordagens para a classificação de atividades. Os resultados preliminares obtidos demonstram que a aquisição do sinal do acelerómetro utilizando um *smartphone* Android, posicionado no bolso frontal direito, demonstra ser capaz de classificar atividades físicas.

As características extraídas foram baseadas tanto no domínio do tempo, como no domínio da frequência e testaram-se vários algoritmos para a classificação de atividades, com base numa metodologia de aprendizagem supervisionada.

Os resultados sugerem que as características no domínio da frequência são as que apresentam os melhores resultados na classificação de atividades. A classificação com ambos os domínios ou apenas com o domínio do tempo reduz a eficácia dos classificadores.

As características no domínio do tempo são classificadas com melhores resultados com o algoritmo Árvore de Decisão e obtêm piores resultados com o classificador K-Nearest Neighbour (KNN). Por outro lado, a classificação de características no domínio da frequência apresenta melhores resultados com os algoritmos K-Nearest Neighbour (KNN) e piores resultados com o Support Vector Machine (SVM).

No estudo sobre as características mais relevantes para a classificação, baseado na penalização pela norma L1, podemos concluir que, no domínio da frequência, as

primeiras 14 bandas de frequência são as mais relevantes para a classificação. No domínio do tempo, as 5 características que se mostraram mais importantes para a classificação foram as seguintes:  $a_z$ ,  $\sigma_x$ ,  $\sigma_y$ ,  $\rho_{y,x}$  e  $\rho_{z,x}$ .

O presente estudo, baseado na utilização do sensor acelerómetro, pretende contribuir ativamente para motivar e monitorizar atividade física e saúde mental da sociedade, atraindo a atenção para o papel fundamental do exercício físico e dos benefícios que tem para a saúde.

A aplicação realizada no âmbito deste projeto classifica as atividades que são efetuadas e, com essa informação, é possível calcular a duração da atividade através do *timestamp* e estimar a energia consumida na atividade. Assim, é possível construir um diário das atividades, permitindo ter uma ideia do nível de atividade num determinado dia, mês ou ano e até verificar a evolução ao longo desses períodos.

Como trabalho futuro, pretendemos aumentar o número de amostras tanto intra-sujeito, como inter-sujeito e explorar formas de determinar a intensidade com que atividade é ou foi realizada, como por exemplo, ritmo baixo, médio ou alto.

# Bibliografia

- [1] Allied Business Intelligence. 45 million windows phone and 20 million blackberry 10 smartphones in active use at year-end; enough to keep developers interested, 2013. URL <https://www.abiresearch.com/press/45-million-windows-phone-and-20-million-blackberry>.
- [2] Jaime E Berlin, Kristi L Storti, and Jennifer S Brach. Using activity monitors to measure physical activity in free-living conditions. *Journal of the American Physical Therapy Association*, 2006.
- [3] M. Canovas C novas, H. Silva, A. Louren o, and A. L. N. Fred. Mobilebit: A framework for mobile interaction recording and display. In *INSTICC International Conf. on Health Informatics - Healthinf*, volume -, pages —, February 2013.
- [4] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.
- [5] R. Duda and P. Hart. *Pattern classification and scene analysis*. John Wiley & Sons Inc, 1973.
- [6] Gary Isaac Wolf. Quantified self, 2013. URL <http://aether.com/quantifiedself>.
- [7] The Economist. Counting every moment, 2012. URL <http://www.economist.com/node/21548493>.

- 
- [8] Wikipedia. Quantified self, 2013. URL [http://en.wikipedia.org/wiki/Quantified\\_Self](http://en.wikipedia.org/wiki/Quantified_Self).
- [9] World Health Organization. Physical inactivity: A global public health problem, 2013. URL [http://www.who.int/dietphysicalactivity/factsheet\\_inactivity/en/](http://www.who.int/dietphysicalactivity/factsheet_inactivity/en/).
- [10] World Health Organization. Physical activity and adults, 2013. URL [http://www.who.int/dietphysicalactivity/factsheet\\_adults/en/index.html](http://www.who.int/dietphysicalactivity/factsheet_adults/en/index.html).
- [11] World Health Organization. What is overweight and obesity?, 2013. URL [http://www.who.int/dietphysicalactivity/childhood\\_what/en/index.html](http://www.who.int/dietphysicalactivity/childhood_what/en/index.html).
- [12] World Health Organization. Obesity and overweight, 2013. URL <http://www.who.int/mediacentre/factsheets/fs311/en/>.
- [13] Global Health Observatory Data Repository. Physical inactivity: Physical inactivity by country, 2008. URL <http://apps.who.int/gho/data/node.main.A893>.
- [14] Inc. Gartner. Worldwide mobile phone sales, 2013. URL [www.gartner.com](http://www.gartner.com).
- [15] Samsung. What is bada, 2013. URL <http://www.bada.com/whatisbada/index.html>.
- [16] Samsung. Samsung developers, 2013. URL <http://developer.samsung.com/android>.
- [17] BlackBerry. Blackberry 10 os, 2013. URL <http://us.blackberry.com/software/smartphones/blackberry-10-os.html>.
- [18] Wikipedia. Mobile operating system, 2013. URL [http://en.wikipedia.org/wiki/Mobile\\_operating\\_system](http://en.wikipedia.org/wiki/Mobile_operating_system).
- [19] Android Developers. Dashboards, 2013. URL <http://developer.android.com/about/dashboards/index.html#OpenGL>.



- [20] Satya Komatineni and Dave MacLean. Introducing the Android Computing Platform Pro Android 4. In *Pro Android 4*, chapter 1, pages 1–22. Apress, Berkeley, CA, 2012. ISBN 978-1-4302-3930-7. doi: 10.1007/978-1-4302-3931-4\\_1. URL [http://dx.doi.org/10.1007/978-1-4302-3931-4\\_1](http://dx.doi.org/10.1007/978-1-4302-3931-4_1).
- [21] Android Developers. Sensors overview, 2013. URL [http://developer.android.com/guide/topics/sensors/sensors\\_overview.html](http://developer.android.com/guide/topics/sensors/sensors_overview.html).
- [22] CHRobotics. An-1008 - sensors for orientation estimation. CHRobotics, October 2012.
- [23] Lígia J. Figueiredo, Ana R. Gafaniz, Gustavo S. Lopes, and Rúben Pereira. Aplicações de acelerómetros. IAS 2007 – Instrumentação e Aquisição de Sinais, Dezembro 2007.
- [24] José Manuel Lema Santos and Pedro Daniel Correia Pires. Monitorização da saída de um acelerómetro bi-axial. Instituto Superior Técnico, Dezembro 2006.
- [25] Android Developers. Sensormanager, 2013. URL <http://developer.android.com/reference/android/hardware/SensorManager.html>.
- [26] Mark Pfluger. Google play: Accelerometer frequency, 2012. URL [https://play.google.com/store/apps/details?id=com.cochibo.accfreq&hl=pt\\_PT](https://play.google.com/store/apps/details?id=com.cochibo.accfreq&hl=pt_PT).
- [27] Tae-Seong Kim. Smart phone-based human activity classification and energy expenditure generation in building environments. *International Symposium on Sustainable Healthy Buildings*, 2012.
- [28] Myong-Woo Lee, Adil Mehmood Khan, and Tae-Seong Kim. A single tri-axial accelerometer-based real-time personal life log system capable of human activity recognition and exercise information generation. 2011.
- [29] Neil Zhao. Full-featured pedometer design realized with 3-axis digital accelerometer. 2010.

- 
- [30] Tom Mikael Ahola. Pedometer for running activity using accelerometer sensors on the wrist. *Medical Equipment Insights 2010*, 2010.
- [31] Andrea Mannini and Angelo M. Sabatini. Machine Learning Methods for Classifying Human Physical Activity from On-Body Accelerometers. *Sensors*, 10(2):1154–1175, February 2010. ISSN 1424-8220. doi: 10.3390/s100201154. URL <http://dx.doi.org/10.3390/s100201154>.
- [32] Myong-Woo Lee, Adil Mehmood Khan, and Tae-Seong Kim. A single tri-axial accelerometer-based real-time personal life log system capable of human activity recognition and exercise information generation. *Personal and Ubiquitous Computing*, 15(8):887–898, 2011. URL <http://dblp.uni-trier.de/db/journals/puc/puc15.html#LeeKK11>.
- [33] Wan-Young Chung, Amit Purwar, and Annapurna Sharma. Frequency domain approach for activity classification using accelerometer. *CoRR*, abs/1107.4417, 2011. URL <http://dblp.uni-trier.de/db/journals/corr/corr1107.html#abs-1107-4417>.
- [34] A. Sharma, A. Purwar, Young-Dong Lee, Young-Sook Lee, and Wan-Young Chung. Frequency based classification of activities using accelerometer data. In *Multisensor Fusion and Integration for Intelligent Systems, 2008. MFI 2008. IEEE International Conference on*, pages 150–153, 2008. doi: 10.1109/MFI.2008.4648056.
- [35] *Introduction to Sports Biomechanics - Analysing Human Movement Patterns*. Routledge, 2007.
- [36] Sarah L. Kozey, Kate Lyden, Cheryl A. Howe, John W. Staudenmayes, and Patty S. Freedson. Accelerometer output and met values of common physical activities. 2010.
- [37] Scott E. Crouter, James R. Churilla, and David R. Bassett Jr. Estimating energy expenditure using accelerometers. *European journal of applied physiology*, 2006.

- 
- [38] Run Keeper. The personal trainer in your pocket., 2013. URL <http://http://runkeeper.com/>.
- [39] Runstar. Everyone can be a runstar, 2013. URL <http://runstar.se/>.
- [40] Nike. Nike+ running app, 2013. URL [http://nikeplus.nike.com/plus/products/gps\\_app/](http://nikeplus.nike.com/plus/products/gps_app/).
- [41] Frank Android Software. Running tracker, 2013. URL <http://www.softmimo.com/>.
- [42] inc. Char Software. Gps walk and run tracker, 2013. URL <http://www.charsoftware.com/>.
- [43] tshalif. Welcome to talos rowing, 2013. URL <http://nargila.org/trac/robostroke>.
- [44] Inc. MapMyFitness. Mapmyrun gps running, 2013. URL <http://www.MapMyRun.com>.
- [45] Fitbit Inc. Fitbit, 2013. URL <http://www.fitbit.com/zip>.
- [46] Inc. BodyMedia. Bodymedia, 2013. URL <http://www.bodymedia.com/>.
- [47] *Beginning Android 4 Application Development*. Wrox, 2012.
- [48] W3Schools. Html5 introduction, 2013. URL [http://www.w3schools.com/html/html5\\_intro.asp](http://www.w3schools.com/html/html5_intro.asp).
- [49] W3Schools. Css introduction, 2013. URL [http://www.w3schools.com/css/css\\_intro.asp](http://www.w3schools.com/css/css_intro.asp).
- [50] Bruce Lawson and Remy Sharp. Introducing html5, 2013. URL <http://introducinghtml5.com/>.
- [51] *The Book of CSS3: A Developer's Guide to the Future of Web Design*. William Pollock, 2011.
- [52] *JavaScript & jQuery: The Missing Manual*. Pogue Press, 2011.

- 
- [53] Travendo. Autobahn websocket, 2013. URL <http://autobahn.ws/>.
- [54] W. Cooper. Java design patterns: a tutorial, 2000.
- [55] Jenifer Tidwell. *Designing Interfaces*. O'Reilly, Beijing, 2006. ISBN 978-0-596-00803-1. URL <http://my.safaribooksonline.com/9780596008031>.
- [56] Travendo. Autobahn android, 2013. URL <http://autobahn.ws/android>.
- [57] Travendo. Autobahn python, 2013. URL <http://autobahn.ws/python>.
- [58] Fette and Melnikov. Rfc(6455): The websocket protocol, December 2011. URL <http://tools.ietf.org/html/rfc6455>.
- [59] Bosch Sensortec. Bma220 data sheet, 2011.
- [60] Jian Liu, Takehito Kanno, Masanori Akashi, Wenxi Chen, Daming Wei, Gengfeng Wu, and Naotoshi Takeda. Patterns of bipedal walking on tri-axial acceleration signals and their use in identifying falling risk of older people. In *The Sixth IEEE International Conference on Computer and Information Technology (CIT'06)*.
- [61] The University of Texas at Dallas. Arts technology, 2013. URL <http://www.utdallas.edu/atec/>.
- [62] Najme Zehra Naqvi, Ashwani Kumar, Aanchal Chauhan, and Kritka Sahni. Step counting using smartphone-based accelerometer. *International Journal on Computer Science and Engineering (IJCSE)*, 2012.
- [63] Ainsworth BE, Haskell WL, Leon AS, Jacobs DR Jr, Montoye HJ, Sallis JF, and Paffenbarger RS Jr. Compendium of physical activities: classification of energy costs of human physical activities. 1993.
- [64] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830, 2011.